Segment-based Interactive Machine Translation at a Character Level

Ángel Navarro[1] and **Miguel Domingo**[1,2] and **Francisco Casacuberta**[1,2]
[1]PRHLT Research Center
Universitat Politècnica de València, Spain
{annamar8, midobal, fcn}@prhlt.upv.es
[2]ValgrAI - Valencian Graduate School and Research Network for Artificial Intelligence,
Camí de Vera s/n, 46022 Valencia, Spain

Abstract

To produce high quality translations, human translators need to review and correct machine translation hypothesis in a process known as post-editing. In order to reduce the human effort of this task, interactive machine translation proposed a collaborative framework in which human and machine work together to generate the translations. Among the many protocols proposed throughout the years, the segment-based one established a paradigm in which the post-editor is allowed to validate correct word sequences from a translation hypothesis and to introduce a word correction to help the system improve the next hypothesis. In this work we propose an extension to this protocol: instead of having to type the complete word correction, the system will complete the user's correction while they are typing. We evaluated our proposal under a simulated environment, achieving a significant reduction of the human effort.

1 Introduction

The machine translation (MT) field has significantly changed over the last few years due to the appearance and application of neural models. Thanks to this emergent technology, researchers have been able to accomplish human parity in several MT-related tasks (Toral, 2020). Thus, in the future we might no longer need human translators to review and correct translations hypothesis from an MT system to achieve high-quality translations. Until this future arrives, human experts need to be involved in the translation process and post-edit the MT system's output in order to get translations of the required high quality.

To alleviate the cost of the post-editing task, interactive machine translation (IMT) proposed a collaborative framework in which human and machine work together to construct the final translation: instead of correcting the complete translation hypothesis, the expert can provide the system with some feedback which it uses to generate a new hypothesis. This process is repeated until the user is satisfied with the system's hypothesis.

Among the different protocols proposed in the literature, we find segment-based IMT (Domingo et al., 2017; Peris et al., 2017). In this paradigm, the user reviews the system's translation hypothesis and can validate sequences of words which they consider to be correct. Then, they make a word correction. The system reacts to this feedback by generating a new hypothesis and, thus, starting a new iteration of the process.

Figure 1 illustrates an iteration of a segment-based IMT session where the user has to translate a sentence from Spanish to English. Given the hypothesis generated by the MT model, the user starts validating a sequence of correctly translated segments and types the word *first* to help the system fulfill the sequence of words between the first two validated segments. The system generates a new hypothesis with the feedback from the validated segments and the word correction. The process that describes the figure is repeated until the hypothesis generated by the system is good enough that the user validates it.

In this work, we propose to extend this protocol so that instead of having to make a word correction, the system generates a new hypothesis as soon as

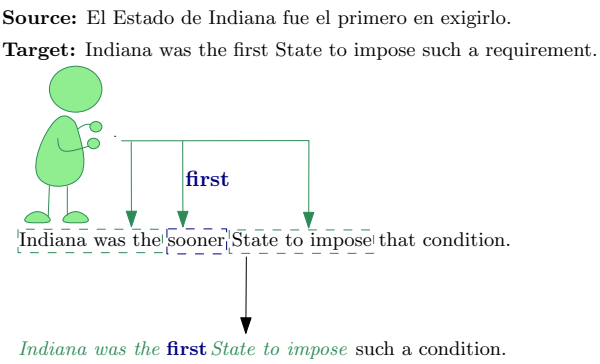

Figure 1: Example of an iteration in the segment-based IMT protocol. The user reviews the system's hypothesis, validating the sequence of words *Indiana was the* and *State to impose* and makes a word correction (**first**). Then, the system generates a new hypothesis that takes into account the user's feedback.

the user starts typing, helping them complete the correction and, thus, reducing even more the typing effort.

2 Related work

Reducing the effort users need to perform during the translation process is a problem that has been in the spotlight of IMT researchers since its paradigm was proposed as an alternative to post-editing (Foster et al., 1997). In this first approach, the user selected a section of the source text and started to type its translation. When the user typed a character, the system displayed a list of possible words that the user might accept or reject. Since then, researchers have studied various approaches to reduce the user effort even more.

Over time, appeared projects such as TransType (Langlais et al., 2000), Matecat (Federico et al., 2014), CasMacat (Alabau et al., 2013), and TranSmart (Huang et al., 2021), whose aim was to create a workbench with an array of innovative features that were not available in other tools at their start. Adding multiple ways to edit a translation and to visualize the information helped to reduce the effort. Among the features that each workbench integrated, they found helpful to use an IMT system to predict either the current word or the rest of the translation.

These projects used the prefix-based protocol introduced by Foster et al. (1997). In this protocol, the user reviews the system's translation hypothesis from left to right, validating one segment from the start of the translation until it finds the first error to correct. The user validates a larger prefix at each iteration, and the system produces an appropriate suffix for completing the translation. The protocol has evolved over the years, presenting advances related to suffix generation (Koehn et al., 2014; Torregrosa et al., 2014; Azadi and Khadivi, 2015), introducing new kinds of interaction (Sanchis-Trilles et al., 2008; Navarro and Casacuberta, 2021b), and visualization of the information with confidence measures (González-Rubio et al., 2010; Navarro and Casacuberta, 2021a).

The segment-based protocol, introduced by Domingo et al. (2017; Peris et al. (2017), has also evolved over the years, applying over it techniques from other MT subfields. Researchers have used reinforcement learning (Lam et al., 2018) and confidence measures (Zhao et al., 2020) to obtain the validated segments and improve segment prediction with text-infilling methods (Xiao et al., 2022).

In this work, we extend the segment-based protocol from typing the whole word to perform a new prediction to only needing to type one character. This same approach has also been studied for the prefix-based protocol (González-Rubio et al., 2013; Santy et al., 2019; Navarro and Casacuberta, 2022).

3 Segment-based IMT

In the segment-based IMT framework, a human translator and an MT system work together to create high-quality translations. This collaboration starts with the system proposing an initial translation hypothesis y_1^I of length I. The user, then, reviews this hypothesis and validates those sequences of words which they consider to be correct ($\tilde{\mathbf{f}}_1, \ldots, \tilde{\mathbf{f}}_N$; where N is the number of non-overlapping validated segments). Next, they are able to merge two consecutive segments $\tilde{\mathbf{f}}_i$, $\tilde{\mathbf{f}}_{i+1}$ into a new one. Finally, they make a word correction—introducing a new one-word validated segment, $\tilde{\mathbf{f}}_i$, which is inserted in $\tilde{\mathbf{f}}_1^N$.

In response to this user feedback, the system generates a sequence of new translation segments $\widehat{\mathbf{g}}_1^N = \widehat{\mathbf{g}}_1, \ldots, \widehat{\mathbf{g}}_N$; where each $\widehat{\mathbf{g}}_n$ is a subsequence of words in the target language. This sequence complements the user's feedback to conform the new hypothesis:

$$\hat{y}_1^I = \tilde{\mathbf{f}}_1, \widehat{\mathbf{g}}_1, \ldots, \tilde{\mathbf{f}}_N, \widehat{\mathbf{g}}_N \tag{1}$$

Peris et al. (2017) formalized the word probability expression for the words belonging to a validated segment $\tilde{\mathbf{f}}_n$ as:

$$p(y_{i_n+i'} \mid y_1^{i_n+i'-1}, x_1^J, f_1^N; \Theta) = \mathbf{y}_{i_n+i'}^\top \mathbf{p}_{i_n+i'}, \quad 1 \leq i' \leq \hat{l}_n \tag{2}$$

where l_n is the size of the non-validated segment generated by the system, which is computed as follows:

$$\hat{l}_n = \arg\max_{0 \leq l_n \leq L} \frac{1}{l_N + 1} \sum_{i'=i_n+1}^{i_n+l_n+1} \log p(y_{i'} \mid y_1^{i'-1}, x_1^J; \Theta) \tag{3}$$

3.1 Character-level segment-based IMT

In this work, we extend the segment-based protocol by allowing a partially typed word $\tilde{\mathbf{f}}'_i$, which the system will complete as part of its prediction. The user can either validate it (replacing the validated segment $\tilde{\mathbf{f}}'_i$ by $\tilde{\mathbf{f}}_i = \tilde{\mathbf{f}}'_i \hat{\mathbf{g}}_i$) or partially validate it—moving the cursor to the desire position—adding $\hat{g}_i^{i+c}$ (were c is the number of new character to validate) into $\tilde{\mathbf{f}}'_i$. Then, if the predicted word has not been validated, the user continues typing. This process is repeated until the word correction is complete, in which case the user shall continue reviewing the new translation hypothesis.

To account for this new feature, we can rewrite Eq. (1) into:

$$\begin{cases} \hat{y}_1^I = \tilde{\mathbf{f}}_1, \hat{\mathbf{g}}_1, \ldots, \tilde{\mathbf{f}}'_i \hat{\mathbf{g}}_i, \ldots, \tilde{\mathbf{f}}_N, \hat{\mathbf{g}}_N \text{ if } \tilde{\mathbf{f}}'_i \in \tilde{\mathbf{f}}_1^N \\ \hat{y}_1^I = \tilde{\mathbf{f}}_1, \hat{\mathbf{g}}_1, \ldots, \tilde{\mathbf{f}}_N, \hat{\mathbf{g}}_N \text{ otherwise} \end{cases} \tag{4}$$

Figure 2 illustrates an iteration of a segment-based IMT session at a character lever where the user must translate a Spanish sentence to English. Starting with the translation generated by the MT model, the user validates a sequence of segments and types the character (*f*) to help the system to complete the space between the two validated segments with the word in its mind (*first*). As soon as they start typing, the system generates a new hypothesis using the feedback provided.

4 Experimental framework

This section presents the details of our experimental session. We start by presenting the evaluation metrics used for assessing our proposal. Then, we describe the corpora used for training our models. After that, we detail the training procedure of our MT systems. Finally, we describe how we performed the user simulation.

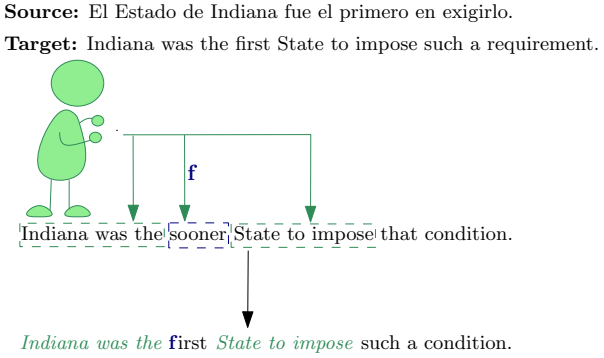

Figure 2: Example of an iteration in the segment-based IMT protocol. The user reviews the system's hypothesis, validating the sequence of words *Indiana was the* and *State to impose* and making a word correction. As soon as they start typing, the system generates a new hypothesis that completes the word correction—taking into account the user's feedback.

4.1 Evaluation metrics

We made use of the following well-known metrics in order to assess our proposal:

Key stroke ratio (KSR) (Tomás and Casacuberta, 2006): measures the number of characters typed by the user, normalized by the number of characters in the final translation.

Mouse action ratio (MAR) (Barrachina et al., 2009): measures the number of mouse actions made by the user, normalized by the number of characters in the final translation.

Keystroke mouse-action ratio (KSMR) (Barrachina et al., 2009): measures the number of characters typed plus the number of mouse actions made by the user, normalized by the number of characters in the final translation.

Additionally, we assessed the initial translation quality of each system using:

Bilingual evaluation understudy (BLEU) (Papineni et al., 2002): computes the geometric average of the modified n-gram precision, multiplied by a brevity factor that penalizes short sentences. In order to ensure consistent BLEU scores, we used *sacreBLEU* (Post, 2018) for computing this metric.

Translation error rate (TER) (Snover et al., 2006): computes the number of word edit

operations (insertion, substitution, deletion and swapping), normalized by the number of words in the final translation. It can be seen as a simplification of the user effort of correcting a translation hypothesis on a classical post-editing scenario.

Finally, we applied approximate randomization testing (ART) (Riezler and Maxwell, 2005)—with $10,000$ repetitions and using a p-value of 0.05—to determine whether two systems presented statistically significance.

4.2 Corpora

Following prior IMT works (Tomás and Casacuberta, 2006; Barrachina et al., 2009), we tested our proposal with four different corpora:

EU[1] (Barrachina et al., 2009): a collection of documents from the *Bulletin of the European Union*.

TED[2] (Federico et al., 2011): a collection of public speeches from a variety of topics.

Xerox (Barrachina et al., 2009): a collection of *Xerox*'s printer manuals.

Europarl (Koehn, 2005): a collection of proceedings from the European Parliament. We used WMT[3][4]'s *news-test2013* and *news-test2015* for De–En's validation and test (respectively), and *news-test2012* and *news-test2013* for Es–En's validation and test (respectively).

Table 1 shows the main features of the corpora.

Table 1: Corpora statistics. K denotes thousands and M millions. $|S|$ stands for number of sentences, $|T|$ for number of tokens and $|V|$ for size of the vocabulary. **Fr** denotes French; **En**, English; **De**, German; and **Es**, Spanish.

		EU		Europarl	
		Fr–En	**De–En**	**De–En**	**Es–En**
Train	$\|S\|$	982.7K	989.2K	1.9M	2.0M
	$\|T\|$	20.7/18.9M	18.0/19.2M	49.8/52.3M	51.6/49.2M
	$\|V\|$	161.4/150.4K	242.5/151.5K	394.6/129.1K	422.6/309.0K
Val.	$\|S\|$	400	400	3000	3003
	$\|T\|$	11.5/10.1K	9.7/10.1K	63.5/64.8K	69.5/63.8K
	$\|V\|$	2.9/2.6K	3.1/2.6K	12.7/9.7K	16.5/14.3K
Test	$\|S\|$	800	800	2169	3000
	$\|T\|$	22.5/20.0	18.8/20.0K	44.1/46.8K	62.0/56.1K
	$\|V\|$	4.5/3.9K	5.0/3.9K	10.0/8.1K	15.2/13.3K

		Xerox		TED
		Es–En	**Fr–En**	**Es–En**
Train	$\|S\|$	55.7K	51.8K	160.2K
	$\|T\|$	0.8/0.7M	0.5/0.6M	3.0/3.2M
	$\|V\|$	16.8/14.0K	24.8/13.7K	89.0/61.7K
Val.	$\|S\|$	1012	964	887
	$\|T\|$	16.0/14.4K	10.7/10.9K	19.2/20.1K
	$\|V\|$	1.8/1.6K	1.7/1.5K	4.1/3.4K
Test	$\|S\|$	1125	984	1570
	$\|T\|$	10.1/8.4K	11.9/12.5K	30.7/32.0K
	$\|V\|$	2.0/1.9K	2.2/1.8K	5.1/3.9K

4.3 Systems

We built our systems using *OpenNMT-py* (Klein et al., 2017). We selected a Transformer architecture (Vaswani et al., 2017) of 6 layers; with all dimensions set to 512 except for the hidden Transformer feed-forward (which was set to 2048); 8 heads of Transformer self-attention; 2 batches of words in a sequence to run the generator on in parallel; a dropout of 0.1; Adam (Kingma and Ba, 2014), using an Adam beta2 of 0.998, a learning rate of 2 and Noam learning rate decay with 8000 warm up steps; label smoothing of 0.1 (Szegedy et al., 2015); beam search with a beam size of 6; and joint byte pair encoding (BPE) (Gage, 1994) applied to all corpora, using $32,000$ merge operations.

4.4 Simulation

Conducting frequent human evaluations at the development stage have a high time and economic costs. Thus, we conducted the evaluation using simulated users whose goal was to generate the translations from the reference.

For the sake of simplicity and without loss of generality, in this simulation we assumed that the user always corrects the leftmost wrong word and that validated word segments must be in the same order as in the reference. This assumption was also made by the authors of the original segment-based protocol (Domingo et al., 2017; Peris et al., 2017).

The simulation starts with the system offering an initial hypothesis. Then, the user reviews it and validates word segments, which are obtained by computing the longest common subsequence (Apostolico and Guerra, 1987) between hypothesis and reference. This has an associated cost of one mouse action for each one-word segment and two for each multi-word segment. After this, the user looks for pairs of consecutive validated segments which could be merged into a single larger segment (i.e., they appear consecutively in the reference but are separated by some words in the hypothesis). If there are, then they merge them, increasing mouse

[1] https://doi.org/10.5281/zenodo.5653096.
[2] https://wit3.fbk.eu/mt.php?release=2013-01.
[3] http://www.statmt.org/wmt12/translation-task.html.
[4] http://www.statmt.org/wmt15/translation-task.html.

Table 2: Results of the character-level segment-based IMT approach in comparison with the word-level approach. All values are reported as percentages. Differences between each approach are statistically significant in all cases. Best results are denoted in bold.

		Translation Quality		Word-level			Character-level		
Corpora	**Language Pair**	**TER [↓]**	**BLEU [↑]**	**KSR [↓]**	**MAR [↓]**	**KSMR [↓]**	**KSR [↓]**	**MAR [↓]**	**KSMR [↓]**
EU	Fr–En	37.4	50.0	19.0	**19.4**	38.4	**7.7**	22.4	**30.1**
	En–Fr	37.5	53.4	17.1	**17.6**	34.7	**6.8**	20.4	**27.2**
	De–En	68.7	26.3	34.5	**27.7**	62.2	**19.5**	31.9	**51.4**
	En–De	52.0	36.9	25.9	**19.9**	45.8	**9.2**	22.7	**31.9**
Europarl	De–En	56.4	24.7	28.7	**27.8**	56.5	**13.4**	32.0	**45.4**
	En–De	60.2	21.9	29.8	**23.3**	53.1	**12.8**	26.5	**39.3**
	Es–En	55.4	26.8	27.0	**27.4**	54.4	**12.1**	31.6	**43.7**
	En–Es	53.0	28.3	27.7	**26.1**	53.8	**12.9**	30.0	**42.9**
Xerox	Es–En	45.7	45.4	25.6	**18.5**	44.1	**16.6**	21.7	**38.3**
	En–Es	45.7	48.2	22.7	**15.8**	38.5	**14.7**	18.6	**33.3**
	Fr–En	56.2	33.0	33.6	**30.3**	63.9	**17.6**	35.1	**52.7**
	En–Fr	56.7	36.3	31.2	**26.2**	57.4	**14.6**	29.6	**44.2**
TED	Es–En	37.1	44.7	20.8	**26.0**	46.8	**10.5**	29.7	**40.2**
	En–Es	42.9	35.8	24.0	**26.3**	50.3	**11.9**	29.8	**41.7**

actions in one if there was a single word between the segments, or two otherwise. Finally, they start correcting the leftmost wrong word. As soon as they start typing, the system reacts to the feedback and generates a new hypothesis which also completes the word correction. If that word is correct, a new iteration of the process starts. If it is not, either the user continues typing or, if part of the predicted word is correct, they move the cursor next to the last correct character (increasing in one the mouse actions) and continue typing the correction (which has a cost of 1 keystroke per character typed). Then, the system reacts to this feedback by generating a new hypothesis. This process is repeated until the hypothesis and the reference are the same.

The software for running these simulations is available together with the implementation of our proposal at GitHub[5].

5 Results

In order to assess our proposal, we evaluated the segment-based IMT protocol at word and character level. We aim to see in the character-level experiments a reduction in the KSR and KSMR due to letting the system try to autocomplete the wrong word instead of typing it manually.

Table 2 shows the experimental results, where the word-level and character-level approaches are compared. The quality of the models in terms of TER and BLEU is included for each experiment to get a grasp of the quality of the initial hypothesis that the simulated users will have to post-edit. In all cases, the character-level method successfully diminishes the typing effort at the expense of a relative small increase of the mouse usage. The KSR is reduced by a factor ranging from 35% to 64%, while MAR values are only increased by a factor of around 15%. This combination of variation on the keystrokes and mouse actions performed results in a reduction of the KSMR by a factor ranging from 13% to 30%.

The translation tasks *Europarl* and *EU* have a higher reduction factor of the KSR. We can deduce that this is due to these corpora having a larger vocabulary, which helps the system to find partially correct words avoiding the worst-case scenario of correcting a word character by character. Moreover, the use of BPE also assists the character level approach, since even if the model does not know the correct word, it is able to predict some of its sub-words correctly.

This high reduction in the KSR is the expected behavior, given that the MT models are good enough to predict correctly the desired word with just a few characters. Even in the worst-case scenario, the system can never correct an error with just a subset of its characters; the KSR maintains the same, as the user needs to type all the characters to rectify the error in both cases. However, working at the character level supposes a minor increment in the MAR because if the next character to correct is not adjacent to the previous one, the user has to move the cursor to the new position. When working at a word level, each word supposes only one mouse action while at a character level each could add multiple mouse actions.

[5] https://github.com/PRHLT/OpenNMT-py/tree/inmt.

Word-level approach

SOURCE: El Estado de Indiana fue el primero en exigirlo.
TARGET: Indiana was the first State to impose such a requirement.

ITER-0	Translation hypothesis	Indiana was the sooner State to impose that condition.
ITER-1	Feedback	*Indiana was the* **first** *State to impose*
	Translation hypothesis	*Indiana was the first State to impose* such a condition.
ITER-2	Feedback	*such a* **requirement**
	Translation hypothesis	*Indiana was the first State to impose such a requirement*.
END	Final translation	*Indiana was the first State to impose such a requirement.*

Post-editing effort: 16 keystrokes and 8 mouse actions.

(a) Word-level segment-based IMT session to translate a sentence from Spanish to English. The process starts with the system offering an initial hypothesis. Then, at iteration 1, the user validates the word segments *Indiana was the* and *State to impose* and makes a word correction (**first**). The system reacts to this feedback by generating a new translation hypothesis. Once more, the user reviews the hypothesis, validating the word segment *such a* and making the word correction **requirement**. Finally, since the next hypothesis is the desired translation, the process ends with the user accepting the translation. Overall, this process has a post-editing effort of 16 keystrokes and 8 mouse actions.

Character-level approach

SOURCE: El Estado de Indiana fue el primero en exigirlo.
TARGET: Indiana was the first State to impose such a requirement.

ITER-0	Translation hypothesis	Indiana was the sooner State to impose that condition.
ITER-1	Feedback	*Indiana was the* **f** *State to impose*
	Word correction	*Indiana was the f*oremost *State to impose* such a condition.
ITER-2	Feedback	*f* **i**
	Word correction	*Indiana was the fi*rst *State to impose* such a condition.
ITER-3	Feedback	*first* *such a* **r**
	Word correction	*Indiana was the first State to impose such a r*equirement.
END	Final translation	*Indiana was the first State to impose such a requirement.*

Post-editing effort: 3 keystrokes and 9 mouse actions.

(b) Character-level segment-based IMT session to translate a sentence from Spanish to English. The process starts with the system offering an initial hypothesis. Then, at iteration 1, the user validates the word segments *Indiana was the* and *State to impose* and starts typing the word correction (**f**). At iteration 2, the system offers a suggestion for this word (*foremost*), which the user declines by continue typing the character **i**. Then, at iteration 3, the system successfully suggests the desired word (*first*). Thus, the user validates it and continues reviewing the new hypothesis (validating the word segment *such a* and typing a new word correction). Finally, since the system's next suggestion is the desired translation, the process ends with the user accepting the translation. Overall, this process has a post-editing effort of 3 keystrokes and 9 mouse actions. This supposes a reduction of 13 keystrokes compared to the word-level approach, at the expenses of increasing the mouse effort by just one additional action.

Figure 3: Example of a segment-based IMT session in which the character-level protocol successfully reduces the post-editing effort.

5.1 Qualitative analysis

Fig. 3 presents and example in which our character-level approach yields significant improvements compared with the word-level approach. At Fig. 3a, the segment-based IMT session starts with the system generating an initial hypothesis which needs to be reviewed and corrected. Then, at iteration 1, the user validates a sequence of segments and types the word *first* to help the system fulfill the sequence of words between the first two validated segments. With the feedback conformed by the validated segments and the word correction, the system generates a new hypothesis. At iteration 2, the user validates new segments and makes a new word correction. This time the translation hypothesis meets the user requirements, so the process ends with the user confirming it at the next iteration. Overall, this process has a post-editing effort of 16 keystrokes and 8 mouse actions.

At Fig. 3b, the character-level segment-based IMT session also starts with the system generating an initial hypothesis that needs to be reviewed and corrected. Then, at iteration 1, the user validates a sequence of segments and types the character (*f*) to help the system complete the sequence of

Word-level approach

SOURCE: Una estrategia republicana para obstaculizar la reelección de Obama
TARGET: A Republican strategy to counter the re-election of Obama

ITER-0	Translation hypothesis	A Republican strategy to hinder the re-election of Obama
ITER-1	Feedback	*A Republican strategy to* **counter** *the re-election of Obama*
	Translation hypothesis	*A Republican strategy to counter the re-election of Obama*
END	Final translation	*A Republican strategy to counter the re-election of Obama*

Post-editing effort: 7 keystrokes and 5 mouse actions.

(a) Word-level segment-based IMT session to translate a sentence from Spanish to English. The process starts with the system offering an initial hypothesis. Then, at iteration 1, the user validates the word segments *A Republican strategy to* and *the re-election of Obama* and makes a word correction (**counter**). The system reacts to this feedback by generating a new translation hypothesis. Finally, since the next hypothesis is the desired translation, the process ends with the user accepting the translation. Overall, this process has a post-editing effort of 7 keystrokes and 5 mouse actions.

Character-level approach

SOURCE: Una estrategia republicana para obstaculizar la reelección de Obama
TARGET: A Republican strategy to counter the re-election of Obama

ITER-0	Translation hypothesis	A Republican strategy to hinder the re-election of Obama
ITER-1	Feedback	*A Republican strategy to* **c** *the re-election of Obama*
	Translation hypothesis	*A Republican strategy to* hinder the *c*hoice of Obama *the re-election of Obama*
ITER-2	Feedback	*co* **o**
	Translation hypothesis	*A Republican strategy to* hinder the *co*nsumption of Obama *the re-election of Obama*
ITER-3	Feedback	*cou* **u**
	Translation hypothesis	*A Republican strategy to* hinder the *cou*rage of Obama *the re-election of Obama*
ITER-4	Feedback	*coun* **n**
	Translation hypothesis	*A Republican strategy to* hinder the *coun*cil of Obama *the re-election of Obama*
ITER-5	Feedback	*count* **t**
	Translation hypothesis	*A Republican strategy to* hinder the *count*ries of Obama *the re-election of Obama*
ITER-6	Feedback	*counte* **e**
	Translation hypothesis	*A Republican strategy to* hinder the *counte*nance of Obama *the re-election of Obama*
ITER-7	Feedback	*counter* **r**
	Translation hypothesis	*A Republican strategy to counter the re-election of Obama*
END	Final translation	*A Republican strategy to counter the re-election of Obama*

Post-editing effort: 7 keystrokes and 6 mouse actions.

(b) Character-level segment-based IMT session to translate a sentence from Spanish to English. In this example, the worst-case scenario happens where the system cannot predict the word the user is trying to correct. The process starts with the system offering an initial hypothesis. Then, at iteration 1, the user validates the word segments *A republican strategy to* and *the re-election of Obama* and starts correcting the word *counter* by typing the character **c**. At the following iterations, the suggestions offered by the system have no relation with the word correction that the user has in mind. Therefore, they must type the whole word. Finally, since the system's next suggestion has included the desired translation, the user merges the validated segments and accepts the translation. Overall, this process has a post-editing effort of 7 keystrokes and 6 mouse actions. Despite being the worst-case scenario, this effort is the same as for the word-level approach (plus an additional mouse action to word completion).

Figure 4: Example of a segment-based IMT session in which the character-level protocol faces the worst-case scenario and obtains the same number of keystrokes as the word-level protocol.

words between the first two validated segments. Immediately, the system reacts and generates a new hypothesis. However, this hypothesis does not correctly complete the word correction the user was aiming for. Thus, at the next iteration, the user will continue typing the next character of the word *first*. This process continues until the user is satisfied with the translation hypothesis. Overall, the process has a post-editing effort of 3 keystrokes and 9 mouse actions. This supposes a reduction of 13 keystrokes compared to the word-level approach, at the expenses of increasing the mouse effort by just one additional action.

Fig. 4 presents an example in which the character-level protocol is unable to correctly complete the word correction, resulting in the same post-editing effort than the word-level approach. At Fig. 4a, the session starts with the system offering an initial hypothesis. Then, at iteration 1, the user reviews it and validates the word segments *A Republican strategy to* and *the re-election of Obama* and makes a word correction (*counter*). The system reacts

to this feedback by generating a new hypothesis which, since is the desired translation, the user accepts. Overall, this process has a post-editing effort of 16 keystrokes and 8 mouse actions.

At Fig. 4b, the session also starts with the system offering an initial hypothesis. At iteration 1, the user reviews it and validates the word segments *A Republican strategy to* and *the re-election of Obama* and starts typing the word correction (*counter*). The system offers a suggestion (choice), which has no relation with the word the user has in mind. Therefore, the user continues typing the correction. The system keeps failing with its suggestions so, finally, the user ends up typing the whole word. The system, then, generates as a new hypothesis the desired translation, and so the process ends with the user accepting it. Overall, this process has a post-editing effort of 7 keystrokes and 6 mouse actions. Despite being the worst-case scenario, this effort is the same as for the word-level approach (plus an additional mouse action to word completion).

6 Conclusions and future work

In this work we have extended the segment-based IMT protocol so that the system also helps the user through the word correction step of the process. Now, instead of having to input the whole word, the system offers suggestions while the user is typing the correction. We assessed our proposal under a simulated environment, observing a significant reduction of the overall human effort.

As a future work we would like to extend this feature by providing the user with a list of suggested words, instead of just auto-completing the word correction with only the most probable one. Additionally, we would like to conduct a user evaluation to better assess the impact of our proposal, taking also into consideration other factors such as time.

Acknowledgements

This work received funding from *Generalitat Valencia* under the program *CIACIF/2021/292* and from *ValgrAI* (*Valencian Graduate School and Research Network for Artificial Intelligence*). It has also been partially supported by grant *PID2021-124719OB-I00* funded by *MCIN/AEI/10.13039/501100011033* and by *European Regional Development Fund* (*ERDF*).

References

[Alabau et al.2013] Alabau, Vicent, Ragnar Bonk, Christian Buck, Michael Carl, Francisco Casacuberta, Mercedes García-Martínez, Jesús González-Rubio, Philipp Koehn, Luis A. Leiva, Bartolomé Mesa-Lao, Daniel Ortiz-Martínez, Hervé Saint-Amand, Germán Sanchis-Trilles, and Chara Tsoukala. 2013. CASMACAT: An open source workbench for advanced computer aided translation. *The Prague Bulletin of Mathematical Linguistics*, 100:101–112.

[Apostolico and Guerra1987] Apostolico, A. and C. Guerra. 1987. The longest common subsequence problem revisited. *Algorithmica*, 2:315–336.

[Azadi and Khadivi2015] Azadi, Fatemeh and Shahram Khadivi. 2015. Improved search strategy for interactive machine translation in computer-asisted translation. In *Proceedings of Machine Translation Summit XV*, pages 319–332.

[Barrachina et al.2009] Barrachina, Sergio, Oliver Bender, Francisco Casacuberta, Jorge Civera, Elsa Cubel, Shahram Khadivi, Antonio Lagarda, Hermann Ney, Jesús Tomás, Enrique Vidal, and Juan-Miguel Vilar. 2009. Statistical approaches to computer-assisted translation. *Computational Linguistics*, 35:3–28.

[Domingo et al.2017] Domingo, Miguel, Álvaro Peris, and Francisco Casacuberta. 2017. Segment-based interactive-predictive machine translation. *Machine Translation*, 31:1–23.

[Federico et al.2011] Federico, Marcello, Luisa Bentivogli, Michael Paul, and Sebastian Stüker. 2011. Overview of the IWSLT 2011 evaluation campaign. In *International Workshop on Spoken Language Translation*, pages 11–27.

[Federico et al.2014] Federico, Marcello, Nicola Bertoldi, Mauro Cettolo, Matteo Negri, Marco Turchi, Marco Trombetti, Alessandro Cattelan, Antonio Farina, Domenico Lupinetti, Andrea Martines, Alberto Massidda, Holger Schwenk, Loïc Barrault, Frédéric Blain, Philipp Koehn, Christian Buck, and Ulrich Germann. 2014. In *The MateCat Tool*, pages 129–132, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.

[Foster et al.1997] Foster, George, Pierre Isabelle, and Pierre Plamondon. 1997. Target-text mediated interactive machine translation. *Machine Translation*, 12:175–194.

[Gage1994] Gage, Philip. 1994. A new algorithm for data compression. *The C Users Journal*, 12(2):23–38.

[González-Rubio et al.2010] González-Rubio, Jesús, Daniel Ortiz-Martínez, and Francisco Casacuberta. 2010. Balancing user effort and translation error in interactive machine translation via confidence measures. In *Proceedings of the Annual Meeting*

of the Association for Computational Linguistics, pages 173–177.

[González-Rubio et al.2013] González-Rubio, Jesús, Daniel Ortiz-Martínez, José-Miguel Benedí, and Francisco Casacuberta. 2013. Interactive machine translation using hierarchical translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 244–254.

[Huang et al.2021] Huang, Guoping, Lemao Liu, Xing Wang, Longyue Wang, Huayang Li, Zhaopeng Tu, Chengyan Huang, and Shuming Shi. 2021. Transmart: A practical interactive machine translation system. *arXiv preprint arXiv:2105.13072*.

[Kingma and Ba2014] Kingma, Diederik P and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[Klein et al.2017] Klein, G., Y. Kim, Y. Deng, J. Senellart, and A. M. Rush. 2017. OpenNMT: Open-Source Toolkit for Neural Machine Translation. In *Proceedings of the Association for Computational Linguistics: System Demonstration*, pages 67–72.

[Koehn et al.2014] Koehn, Philipp, Chara Tsoukala, and Herve Saint-Amand. 2014. Refinements to interactive translation prediction based on search graphs. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 574–578.

[Koehn2005] Koehn, Philipp. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of the Machine Translation Summit*, pages 79–86.

[Lam et al.2018] Lam, Tsz Kin, Julia Kreutzer, and Stefan Riezler. 2018. A reinforcement learning approach to interactive-predictive neural machine translation. *arXiv preprint arXiv:1805.01553*.

[Langlais et al.2000] Langlais, Philippe, George Foster, and Guy Lapalme. 2000. In *TransType: a Computer-Aided Translation Typing System*, pages 46–51.

[Navarro and Casacuberta2021a] Navarro, Angel and Francisco Casacuberta. 2021a. Confidence measures for interactive neural machine translation. In *Proceedings of the IberSPEECH conference*, pages 195–199.

[Navarro and Casacuberta2021b] Navarro, Angel and Francisco Casacuberta. 2021b. Introducing mouse actions into interactive-predictive neural machine translation. In *Proceedings of the Machine Translation Summit*. In press.

[Navarro and Casacuberta2022] Navarro, Ángel and Francisco Casacuberta. 2022. On the use of mouse actions at the character level. *Information*, 13(6):294.

[Papineni et al.2002] Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

[Peris et al.2017] Peris, Álvaro, Miguel Domingo, and Francisco Casacuberta. 2017. Interactive neural machine translation. *Computer Speech & Language*, 45:201–220.

[Post2018] Post, Matt. 2018. A call for clarity in reporting bleu scores. In *Proceedings of the Third Conference on Machine Translation*, pages 186–191.

[Riezler and Maxwell2005] Riezler, Stefan and John T Maxwell. 2005. On some pitfalls in automatic evaluation and significance testing for mt. In *Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 57–64.

[Sanchis-Trilles et al.2008] Sanchis-Trilles, Germán, Daniel Ortiz-Martínez, Jorge Civera, Francisco Casacuberta, Enrique Vidal, and Hieu Hoang. 2008. Improving interactive machine translation via mouse actions. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 485–494.

[Santy et al.2019] Santy, Sebastin, Sandipan Dandapat, Monojit Choudhury, and Kalika Bali. 2019. INMT: Interactive neural machine translation prediction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 103–108.

[Snover et al.2006] Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the Association for Machine Translation in the Americas*, pages 223–231.

[Szegedy et al.2015] Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9.

[Tomás and Casacuberta2006] Tomás, Jesús and Francisco Casacuberta. 2006. Statistical phrase-based models for interactive computer-assisted translation. In *Proceedings of the International Conference on Computational Linguistics/Association for Computational Linguistics*, pages 835–841.

[Toral2020] Toral, Antonio. 2020. Reassessing claims of human parity and super-human performance in machine translation at wmt 2019. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 185–194.

[Torregrosa et al.2014] Torregrosa, Daniel, Mikel L. Forcada, and Juan Antonio Pérez-Ortiz. 2014. An open-source web-based tool for resource-agnostic interactive translation prediction. *Prague Bulletin of Mathematical Linguistics*, 102:69–80.

[Vaswani et al.2017] Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

[Xiao et al.2022] Xiao, Yanling, Lemao Liu, Guoping Huang, Qu Cui, Shujian Huang, Shuming Shi, and Jiajun Chen. 2022. Bitiimt: a bilingual text-infilling method for interactive machine translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1958–1969.

[Zhao et al.2020] Zhao, Tianxiang, Lemao Liu, Guoping Huang, Huayang Li, Yingling Liu, Liu GuiQuan, and Shuming Shi. 2020. Balancing quality and human involvement: An effective approach to interactive neural machine translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 9660–9667.