



HAL
open science

BiSync: A Bilingual Editor for Synchronized Monolingual Texts

Josep Crego, Jitao Xu, François Yvon

► **To cite this version:**

Josep Crego, Jitao Xu, François Yvon. BiSync: A Bilingual Editor for Synchronized Monolingual Texts. The 61st Annual Meeting of the Association for Computational Linguistics, ACL, Jul 2023, Toronto, Canada. pp.369-376. hal-04163029

HAL Id: hal-04163029

<https://hal.science/hal-04163029>

Submitted on 17 Jul 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - ShareAlike 4.0 International License

BiSync: A Bilingual Editor for Synchronized Monolingual Texts

Josep Crego[‡]

Jitao Xu[†]

François Yvon[§]

[‡]SYSTRAN, 5 rue Feydeau, 75002 Paris, France

[†]NetEase YouDao, Beijing, China

[§]Sorbonne Université, CNRS, ISIR, F-75005 Paris, France

firstname.lastname@{[‡]systrangroup.com, [§]cnrs.fr} xujt01@rd.netease.com

Abstract

In our globalized world, a growing number of situations arise where people are required to communicate in one or several foreign languages. In the case of written communication, users with a good command of a foreign language may find assistance from computer-aided translation (CAT) technologies. These technologies often allow users to access external resources, such as dictionaries, terminologies or bilingual concordancers, thereby interrupting and considerably hindering the writing process. In addition, CAT systems assume that the source sentence is fixed and also restrict the possible changes on the target side. In order to make the writing process smoother, we present BiSync, a bilingual writing assistant that allows users to freely compose text in two languages, while maintaining the two monolingual texts synchronized. We also include additional functionalities, such as the display of alternative prefix translations and paraphrases, which are intended to facilitate the authoring of texts. We detail the model architecture used for synchronization and evaluate the resulting tool, showing that high accuracy can be attained with limited computational resources. The interface and models are publicly available at <https://github.com/jmcrego/BiSync> and a demonstration video can be watched on YouTube.

1 Introduction

In today’s globalized world, there is an ever-growing demand for multilingual communication. To give just a few examples, researchers from different countries often write articles in English, international companies with foreign subsidiaries need to produce documents in multiple languages, research institutions communicate in both English and the local language, etc. However, for many people, writing in a foreign language (L2) other than their native language (L1) is not an easy task.

With the significant advances in machine translation (MT) in the recent years, in particular due to the tangible progress in neural machine translation (NMT, Bahdanau et al., 2015; Vaswani et al., 2017), MT systems are delivering usable translations in an increasing number of situations. However, it is not yet realistic to rely on NMT technologies to produce high quality documents, as current state-of-the-art systems have not reached the level where they could produce error-free translations. Also, fully automatic translation does not enable users to precisely control the output translations (e.g. with respect to style, formality, or term use). Therefore, users with a good command of L2, but not at a professional level, can find help from existing computer-assisted language learning tools or computer-assisted translation (CAT) systems. These tools typically provide access to external resources such as dictionaries, terminologies, or bilingual concordancers (Bourdaillet et al., 2011) to help with writing. However, consulting external resources causes an interruption in the writing process due to the initiation of another cognitive activity, even when writing in L1 (Leijten et al., 2014). Furthermore, L2 users tend to rely on L1 (Wolfersberger, 2003) to prevent a breakdown in the writing process (Cumming, 1989). To this end, several studies have focused on developing MT systems that ease the writing of texts in L2 (Koehn, 2010; Huang et al., 2012; Venkatapathy and Mirkin, 2012; Chen et al., 2012; Liu et al., 2016).

However, existing studies often assume that users can decide whether the provided L2 texts precisely convey what they want to express. Yet, for users who are not at a professional level, the evaluation of L2 texts may not be so easy. To mitigate this issue, researchers have also explored round-trip translation (RTT), which translates the MT output in L2 back to L1 in order to evaluate the quality of L2 translation (Moon et al., 2020). Such studies suggest that it is then helpful to augment L2 writing



Figure 1: User interface of our online bilingual editing system. Users can freely choose the language in which they compose and alternate between text entry boxes. The system automatically keeps the other box in sync.

with the display of the corresponding synchronized version of the L1 text, in order to help users verify their composition. In such settings, users can obtain synchronized texts in two languages, while only making an effort to only compose in one.

A bilingual writing assistant system should allow users to write freely in both languages and always provide synchronized monolingual texts in the two languages. However, existing systems do not support both functionalities simultaneously. The system proposed by Chen et al. (2012) enables free composition in two languages, but only displays the final texts in L2. Commercial MT systems like Google,¹ DeepL² and SYSTRAN³ always display texts in both languages, but users can only modify the source side, while the target side is predicted by the system and is either locked or can only be modified with alternative translations proposed by the system. CAT tools, on the contrary, assume the source sentence is fixed and only allow edits on the target side.


In this paper, we present BiSync, a bilingual writing assistant aiming to extend commercial MT systems by letting users freely alternate between two languages, changing the input text box at their will, with the goal of authoring two equally good and semantically equivalent versions of the text.

2 BiSync Text Editor

In this work, we are interested in a writing scenario that broadens the existing commercial online translation systems. We assume that the user wants to edit or revise a text simultaneously in two languages. See Figure 1 for a snapshot of our BiSync assistant. Once the text is initially edited in one language, the other language is automatically synchronized so that the two entry boxes always contain mutual translations. In an iterative process, and until the user is fully satisfied with the content, texts are revised in either language, triggering automatic synchronizations to ensure that both texts remain

mutual translations. The next paragraphs detail the most important features of our online BiSync text editor.

Bidirectional Translations The editor allows users to edit both text boxes at their will. This means that the underlying synchronization model has to perform translations in both directions, as the role of the source and target texts are not fixed and can change over time.

Synchronization This is the most important feature of the editor. It ensures that the two texts are always translations of each other. As soon as one text box is modified, BiSync synchronizes the other box. To enhance the user experience, the system waits a few seconds (delay) before the synchronization takes place. When a text box is modified, the system prevents the second box from being edited until the synchronization has been completed. Users can also disable the synchronization process, using the "freeze" button (). In this case, the frozen text will not be synchronized (modified). Changes are only allowed in the unfrozen text box. This is the standard *modus operandi* of most commercial translation systems that consider the input text as frozen, allowing only a limited number of edits in the translation box.

Prefix Alternatives The editor can also provide several translation alternatives for a given sentence prefix. When users click just before a word w in a synchronized sentence pair, the system displays the most likely alternatives that can complete the translation starting from the word w in a drop-down menu. Figure 2 (bottom) illustrates this functionality, where translation alternatives are displayed after the prefix "Je rentre", in the context of the English sentence "I'm going home because I'm tired". In the example in Figure 2 (bottom), the user clicked right before the French word "à".

Paraphrase Alternatives Another important feature of our BiSync editor is the ability to propose edits for sequences of words at arbitrary positions in both text boxes. Figure 2 (top) illustrates

¹<https://translate.google.com/>

²<https://www.deepl.com/translator>

³<https://www.systran.net/en/translate/>

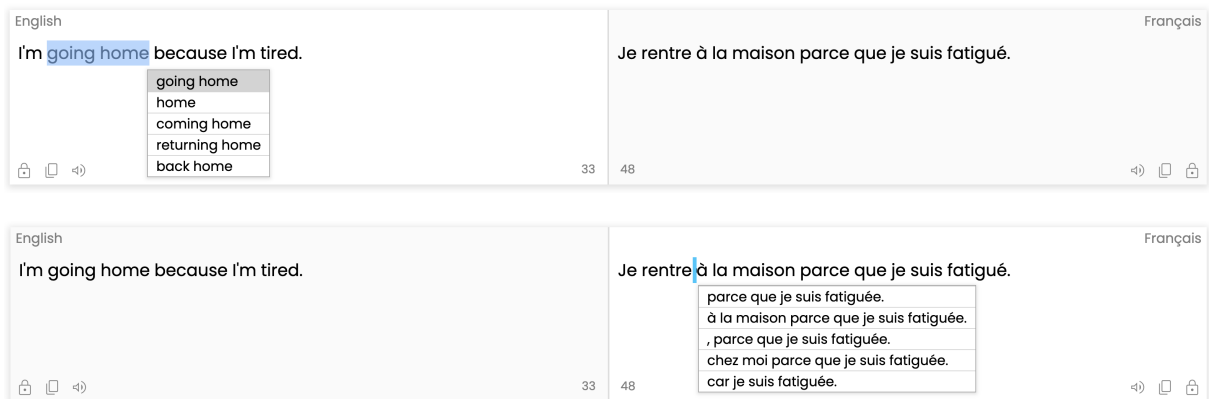


Figure 2: BiSync editor displaying paraphrases (top) and translation alternatives for a given prefix (bottom).

this scenario, where paraphrases for the English segment "going home" are displayed in the context "I'm ... because I'm tired" and given the French sentence "Je rentre à la maison parce que je suis fatigué". Such alternatives are triggered through the selection of word sequences in either text box.

Other Features Like most online translation systems, BiSync has a "listen" button that uses a text-to-speech synthesizer to read the content in the text box, a "copy" button that copies the content to the clipboard. It also displays the number of characters written in each text box. Figures 1 and 2 illustrate these features.

Settings The "gear" button (⚙️) pops up several system parameters that can be configured by users: The "IP" and "Port" fields identify the address where the BiSync model is launched and waits for translation requests. The "Languages" field indicates the pair of languages that the model understands and is able to translate. "Alternatives" denotes the number of hypotheses that the model generates and that are displayed by the system. "Delay" sets the number of seconds the system waits after an edit takes place before starting the synchronization. The countdown is reset each time a revision is made. Figure 3 displays BiSync settings with default parameters.

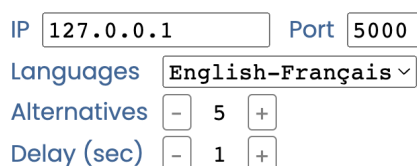


Figure 3: Default BiSync settings.

3 Under the Hood: the BiSync Model

Given the features that we would like to offer in the BiSync text editor, we are interested in an end-to-end model producing: (a) translations in both directions; (b) translations from scratch, with only one text box filled in; (c) updates of existing translations, so that small changes in one text box result in small updates in the other text box; (d) paraphrases and alternatives in context.

We consider a pair of parallel sentences (f , e) and a sentence f' as an update of f . The objective is to generate the sentence e' that is parallel to f' while remaining as close as possible to e . Three types of update are distinguished. Figure 4 displays an example for each update type:

- **Insertion:** adding one or more consecutive words at any position in f ;
- **Deletion:** removing one or more consecutive words at any position in f ;
- **Substitution:** replacing one or more consecutive words at any position in f by one or more consecutive words.

Note that in practice, training such models requires triplets (f' , e , e'), as sentences f are not used by the models studied in this work.

Inspired by Xiao et al. (2022), we integrate several control tokens into the source-side of training examples of a standard NMT model to obtain the desired results. This approach is straightforward and does not require to modify NMT architectures or decoding algorithms. Therefore, our integration of control tokens is model-agnostic and can be applied to any NMT architecture. Several tokens are used to indicate the target language ($\langle en \rangle$, $\langle fr \rangle$)

<i>Update</i>	f	f'	e	e'
<i>Ins</i>	The cat	The white cat	Le chat	Le chat blanc
<i>Del</i>	The cat is white	The white cat	Le chat est blanc	Le chat blanc
<i>Sub</i>	The black cat	The white cat	Le chat noir	Le chat blanc

Figure 4: Source sentences **f** when updated (**f'**) by means of insertion (*Ins*), deletion (*Del*) and substitution (*Sub*) and their corresponding translations (**e** and **e'**). Source sentences **f** are not employed by the models of this work.

and the update type (<ins>, , <sub>). A special token <gap> indicates a sequence of masked tokens used to generate paraphrases.

Training examples are build following the next three patterns:

$$\boxed{\mathbf{f}' \langle \text{lang} \rangle \mid \mathbf{e}'}$$

The first pattern refers to a regular translation task (TRN), and is used when translating from scratch, without any initial sentence pair (**f**, **e**), as in the following example:

$$\textit{The white cat} \langle \text{fr} \rangle \mid \textit{Le chat blanc}$$

where only the target language tag is appended to the end of the source sentence.

$$\boxed{\mathbf{f}' \langle \text{lang} \rangle \mathbf{e} \langle \text{update} \rangle \mid \mathbf{e}'}$$

The second pattern corresponds to an update task (INS, DEL or SUB) to be used for resynchronizing an initial sentence pair (**f**, **e**) after changing **f** into **f'**, as shown in the following examples:

$$\begin{array}{l|l} \textit{The white cat} \langle \text{fr} \rangle \textit{Le chat} \langle \text{ins} \rangle & \textit{Le chat blanc} \\ \textit{The white cat} \langle \text{fr} \rangle \textit{Le chat est blanc} \langle \text{del} \rangle & \textit{Le chat blanc} \\ \textit{The white cat} \langle \text{fr} \rangle \textit{Le chat noir} \langle \text{sub} \rangle & \textit{Le chat blanc} \end{array}$$

where the edited source sentence $\mathbf{f}' = [\textit{The white cat}]$ is followed by the target language tag <fr>, the initial target sentence **e**, and a tag indicating the edit type that updates the initial source sentence **f**.

$$\boxed{\mathbf{f} \langle \text{lang} \rangle \mathbf{e}_g \mid \mathbf{e}_G}$$

The third pattern corresponds to a bilingual text infilling task (BTI, Xiao et al., 2022). The model is trained to predict the tokens masked in a target sentence \mathbf{e}_g in the context of the source sentence **f**:

$$\textit{The white cat} \langle \text{fr} \rangle \textit{Le} \langle \text{gap} \rangle \textit{blanc} \mid \textit{chat}$$

where $\mathbf{e}_g = [\textit{Le} \langle \text{gap} \rangle \textit{blanc}]$ is the target sentence with missing tokens to be predicted. The model only generates the masked tokens $\mathbf{e}_G = [\textit{chat}]$.

3.1 Synthetic Data Generation

While large amounts of parallel bilingual data (**f'**, **e'**) exist for many language pairs, the triplets required to train our model are hardly available. We

therefore study ways to generate synthetic triplets of example (**f'**, **e**, **e'**) from parallel data (**f'**, **e'**) for each type of task (INS, DEL, SUB and BTI) introduced above.

Insertion We build examples of initial translations **e** for INS by randomly dropping a segment from the updated target **e'**. The length of the removed segment is also randomly sampled with a maximum length of 5 tokens. We also impose that the overall ratio of removed segment does not exceed 0.5 of the length of **e'**.

Deletion Simulating deletions requires the initial translation **e** to be an extension of the updated target **e'**. To obtain extensions, we employ a NMT model enhanced to fill in gaps (fill-in-gaps). This model is a regular encoder-decoder Transformer model trained with a balanced number of regular parallel examples (TRN) and paraphrase examples (BTI) as detailed in the previous section. We extend training examples (**f'**, **e'**) with a <gap> token inserted in a random position in **e'** and use fill-in-gaps to decode these training sentences, as proposed in (Xu et al., 2022). In response, the model predicts tokens that best fill the gap. For instance:

$$\textit{The white cat} \langle \text{fr} \rangle \textit{Le chat} \langle \text{gap} \rangle \textit{blanc} \rightsquigarrow \textit{est}$$

the target extension is therefore $\mathbf{e} = [\textit{Le chat est blanc}]$.

Substitution Similar to deletion, substitution examples are obtained using the same fill-in-gaps model. A random segment is masked from **e'**, which is then filled by the model. In inference, the model computes an *n*-best list of substitutions for the mask, and we select the most likely sequence that is not identical to the masked segment. For instance:

$$\textit{The white cat} \langle \text{fr} \rangle \textit{Le chat} \langle \text{gap} \rangle \rightsquigarrow [\textit{blanc}; \textit{bleu}; \textit{clair}; \textit{blanche}; \dots]$$

the target substitution is $\mathbf{e} = [\textit{Le chat bleu}]$.

Note that extensions and substitutions generated by fill-in-gaps may be ungrammatical. For instance, the proposed substitution $e = [Le\ chat\ blanche]$ has a gender agreement error. The correct adjective should be "blanc" (masculine) instead of "blanche" (feminine). This way, the model always learns to produce grammatical e' sentences parallel to f' .

Paraphrase Given sentence pairs (f, e) , we generate e_g by masking a random segment from the initial target sentence e . The length of the masked segment is also randomly sampled with a maximum length of 5 tokens. The target side of these examples (e_G) only contains the masked token(s).

4 Experiments

4.1 Datasets

To train our English-French (En-Fr) models we use the official WMT14 En-Fr corpora⁴ as well as the OpenSubtitles corpus⁵ (Lison and Tiedemann, 2016). A very light preprocessing step is performed to normalize punctuation and to discard examples exceeding a length ratio 1.5 and a limit of [1, 250] measured in words. Statistics of each corpus is reported in Table 1.

<i>Corpora</i>	<i>#Sentences</i>
Europarl v7	2,007,723
Commoncrawl	3,244,152
UN	12,886,831
News Commentary	183,251
Giga French-English	22,520,376
Open Subtitles v18	57,123,540
Total	97,965,873

Table 1: Statistics of training corpora.

For testing, we used the official newstest2014 En-Fr test set made available for the same WMT14 shared task containing 3,003 sentences. All our data is tokenized using OpenNMT tokenizer.⁶ We learn a joint Byte Pair Encoding (Sennrich et al., 2016) over English and French training data with 32k merge operations.

The training corpora used for learning our model consist of well-formed sentences. Most sentences start with a capital letter and end with a punctuation

⁴<https://www.statmt.org/wmt14>

⁵<https://opus.nlpl.eu/OpenSubtitles-v2018.php>

⁶<https://github.com/OpenNMT/Tokenizer>

mark. However, our BiSync editor expects also incomplete sentences, when synchronization occurs before completing the text. To train our model to handle this type of sentences, we lowercase the first character of sentences and remove ending punctuation in both source and target examples with a probability set to 0.05.

4.2 Experimental Settings

Our BiSync model is built using the Transformer architecture (Vaswani et al., 2017) implemented in OpenNMT-tf⁷ (Klein et al., 2017). More precisely, we use the following set-up: embedding size: 1,024; number of layers: 6; number of heads: 16; feedforward layer size: 4,096; and dropout rate: 0.1. We share parameters for input and output embedding layers (Press and Wolf, 2017). We train our models using Noam schedule (Vaswani et al., 2017) with 4,000 warm-up iterations. Training is performed over a single V100 GPU during 500k steps with a batch size of 16,384 tokens per step. We apply label smoothing to the cross-entropy loss with a rate of 0.1. Resulting models are built after averaging the last ten saved checkpoints of the training process. For inference, we use CTranslate2.⁸ It implements custom run-time with many performance optimization techniques to accelerate decoding execution and reduce memory usage of models on CPU and GPU. We also evaluate our model with weight quantization using 8-bit integer (int8) precision, thus reducing model size and accelerating execution compared to the default 32-bit float (float) precision.

5 Evaluation

We evaluate the performance of our synchronization model BiSync compared to a baseline translation model with exactly the same characteristics but trained only on the TRN task over bidirectional parallel data (base). We report performance with BLEU score (Papineni et al., 2002) implemented in SacreBLEU⁹ (Post, 2018) over concatenated En-Fr and Fr-En test sets. For tasks requiring an initial target e , we synthesize e from (f', e') pairs following the same procedures used for generating the training set (see details in Section 2).

Table 2 reports BLEU scores for our two systems on all tasks. The base system is only trained to

⁷<https://github.com/OpenNMT/OpenNMT-tf>

⁸<https://github.com/OpenNMT/CTranslate2>

⁹<https://github.com/mjpost/sacrebleu>. Signature: nrefs:1lcase:mixedlfff:noltok:13alsmooth:expl version:2.0.0

perform regular translations (TRN) for which it obtains a BLEU score of 36.0, outperforming BiSync, which is trained to perform all tasks. This difference can be explained by the fact that BiSync must learn a larger number of tasks than base. When performing INS, DEL and SUB tasks, BiSync vastly outperforms the results of TRN task as it makes good use of the initial translation e . When we use BiSync to generate paraphrases of an initial input (BTI), we obtain a higher BLEU score of 42.6 than the regular translation task (TRN, 34.9). This demonstrates the positive impact of using target side context for paraphrasing.

BLEU	TRN	INS	DEL	SUB	BTI
base	36.0	-	-	-	-
BiSync	34.9	87.9	95.5	78.2	42.6

Table 2: BLEU scores over concatenated En-Fr and Fr-En test sets for all tasks.

Next, we evaluate the ability of our BiSync model to remain close to an initial translation when performing synchronization. Note that for a pleasant editing experience, synchronization should introduce only a minimum number of changes. Otherwise, despite re-establishing synchronization, additional changes may result in losing updates previously performed by the user. To evaluate this capability of our model, we take an initial translation (f, e) and introduce a synthetic update (say f') as detailed in Section 3.1. This update leads to a new synchronization that transforms e into e' . We would like e' to remain as close as possible to e . Table 3 reports TER scores (Snoover et al., 2006) between e and e' computed by SacreBLEU.¹⁰ These results indicate that BiSync produces synchronizations significantly closer to initial translations than those produced by base. This also confirms the findings of Xu et al. (2022).

TER ↓	INS	DEL	SUB
base	36.5	43.4	34.9
BiSync	3.3	5.5	5.0

Table 3: TER scores between e and e' issued from different update types. En-Fr and Fr-En test sets are concatenated.

Finally, Table 4 reports inference efficiency for our BiSync model using CTranslate2. We indi-

¹⁰Signature: nrefs:1|case:l|tok:tercom|norm:nolpunct:yes|asian:nolversion:2.0.0

cate differences in model (*Size* and *Speed*) for different quantization, device, batch size and number of threads. Lower memory requirement and higher inference speed can be obtained by using quantization set to int-8 for both GPU and CPU devices, in contrast to float-32. When running on CPUs, additional speedup is obtained with multithreading. Comparable BLEU scores are obtained in all configurations. Note that for the tool presented in this paper, we must retain single batch size and single thread results (bold figures), since synchronization requests are produced for isolated sentences. Therefore, they cannot take advantage of using multiple threads and large batches.

<i>Quant</i>	<i>Dev</i>	<i>BS</i>	<i>Threads</i>	<i>Size</i>	<i>Speed</i>
float	GPU	64	1	232M	10,267
	GPU	1	1		650
	CPU ¹	64	8 × 4		2,007
	CPU ²	1	1		48
int8	GPU	64	1	59M	12,918
	GPU	1	1		738
	CPU ¹	64	8 × 4		2,666
	CPU ²	1	1		118

Table 4: Inference *Speed* and model *Size* when decoding test sets with several settings: quantization (*Quant*), device (*Dev*), batch size (*BS*) and number of *Threads*. Decoding beam size is set to 3. *Speed* is measured in tokens/second. GPU is a single V100 GPU with 32Gb memory. CPU¹ has 32 cores with 86Gb memory and CPU² is an Intel i7-10850H with 32Gb memory.

6 Conclusion and Further Work

In this paper, we presented BiSync, a bilingual writing assistant system that allows users to freely compose text in two languages while always displaying the two monolingual texts synchronized with the goal of authoring two equally good versions of the text. Whenever users make revisions on either text box, BiSync takes into account the initial translation and reduces the number of changes needed in the other text box as much as possible to restore parallelism. BiSync also assists in the writing process by suggesting alternative reformulations for word sequences or alternative translations based on given prefixes. The synchronization process applies several performance optimization techniques to accelerate inference and reduce the memory usage with no accuracy loss, making BiSync usable even on machines with limited computing power.

In the future, we plan to equip BiSync with a

grammatical error prediction model and a better handling of prior revisions: the aim is to enable finer-grained distinction between parts that the system should modify and parts that have already been fixed or that should remain unchanged. Last, we would like to perform user studies to assess the division of labor between users and BiSync in an actual bilingual writing scenario.

Acknowledgements

We would like to thank the anonymous reviewers for their valuable suggestions and comments. This work was performed while the last two authors were affiliated with LISN, CNRS (Orsay). Jitao Xu has been partly funded by SYSTRAN and by a grant (Transwrite) from the Région Ile-de-France. He was granted access to the HPC resources of IDRIS under the allocation 2022-[AD011011580R2] made by GENCI.

References

- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. [Neural machine translation by jointly learning to align and translate](#). In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.
- Julien Bourdaillet, Stéphane Huet, Philippe Langlais, and Guy Lapalme. 2011. [TransSearch: from a bilingual concordancer to a translation finder](#). *Machine Translation*, 24(3):241.
- Mei-Hua Chen, Shih-Ting Huang, Hung-Ting Hsieh, Ting-Hui Kao, and Jason S. Chang. 2012. [FLOW: A first-language-oriented writing assistant system](#). In *Proceedings of the ACL 2012 System Demonstrations*, pages 157–162, Jeju Island, Korea. Association for Computational Linguistics.
- Alister Cumming. 1989. [Writing expertise and second-language proficiency*](#). *Language Learning*, 39(1):81–135.
- Chung-chi Huang, Ping-che Yang, Keh-jiann Chen, and Jason S. Chang. 2012. [Transahead: A computer-assisted translation and writing tool](#). In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 352–356, Montréal, Canada. Association for Computational Linguistics.
- Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. [OpenNMT: Open-source toolkit for neural machine translation](#). In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada. Association for Computational Linguistics.
- Philipp Koehn. 2010. [Enabling monolingual translators: Post-editing vs. options](#). In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 537–545, Los Angeles, California. Association for Computational Linguistics.
- Mariëlle Leijten, Luuk Van Waes, Karen Schriver, and John R. Hayes. 2014. [Writing in the workplace: Constructing documents using multiple digital sources](#). *Journal of Writing Research*, 5(3):285–337.
- Pierre Lison and Jörg Tiedemann. 2016. [OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles](#). In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 923–929, Portorož, Slovenia. European Language Resources Association (ELRA).
- Lemao Liu, Masao Utiyama, Andrew Finch, and Eiichiro Sumita. 2016. [Agreement on target-bidirectional neural machine translation](#). In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 411–416, San Diego, California. Association for Computational Linguistics.
- Jihyung Moon, Hyunchang Cho, and Eunjeong L. Park. 2020. [Revisiting round-trip translation for quality estimation](#). In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 91–104, Lisboa, Portugal. European Association for Machine Translation.
- 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 of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.
- Matt Post. 2018. [A call for clarity in reporting BLEU scores](#). In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.
- Ofir Press and Lior Wolf. 2017. [Using the output embedding to improve language models](#). In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 157–163, Valencia, Spain. Association for Computational Linguistics.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. [Neural machine translation of rare words with subword units](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

- Matthew Snover, Bonnie Dorr, Rich Schwartz, Linea Micciulla, and John Makhoul. 2006. [A study of translation edit rate with targeted human annotation](#). In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231, Cambridge, Massachusetts, USA. Association for Machine Translation in the Americas.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.
- Sriram Venkatapathy and Shachar Mirkin. 2012. [An SMT-driven authoring tool](#). In *Proceedings of COLING 2012: Demonstration Papers*, pages 459–466, Mumbai, India. The COLING 2012 Organizing Committee.
- Mark Wolfersberger. 2003. [L1 to L2 writing process and strategy transfer: A look at lower proficiency writers](#). *TESL-EJ*, 7(2):1–12.
- Yanling Xiao, Lemaoy 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, Dublin, Ireland. Association for Computational Linguistics.
- Jitao Xu, Josep Crego, and François Yvon. 2022. [Bilingual synchronization: Restoring translational relationships with editing operations](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8016–8030, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.