PRHLT's Submission to WLAC 2023

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

This paper describes our submission to the *Word-Level AutoCompletion* shared task of WMT23. We participated in the English–German and German–English categories. We extended our last year segment-based interactive machine translation approach to address its weakness when no context is available. Additionally, we fine-tune the pre-trained mT5 large language model to be used for autocompletion.

1 Introduction

Despite its improvement in recent years with the emergence of neural machine translation (NMT), machine translation (MT) still cannot assure high-quality translations for all tasks (Toral, 2020). As a consequence, it is critical for professional translators to manually validate the translations generated by the NMT system for those scenarios with rigorous translation quality requirements. Computer-aided translation (CAT) tools emerged to improve the validation and editing process carried out by translators. With the aim of reducing the human effort of correcting the automatic translations, researchers approached CAT tools from many directions. Among CAT tools such as translation memory (Zetzche, 2007), augmented translation (Lommel, 2018) and terminology management (Verplaetse and Lambrechts, 2019); we can find autocompletion tools, which help professional translators by providing new partial translations according to the validated parts they have supplied to the system.

Word level autocompletion (WLAC) (Lin et al., 2021) was introduced as a shared task in WMT22 (Casacuberta et al., 2022). Its aim is to complete a target word given a source sentence, a sequence of characters typed by the human translator and a translation context. Four types of context are possible:

Zero-contex: no context is given.

Suffix: a sequence of translated words located after the word to autocomplete.

Prefix: a sequence of translated words located prior to the word to autocomplete.

Bi-contex: A combination of the *suffix* and the *prefix* type. That is, there is a sequence of translated words located after the word to autocomplete, and a sequence of translated words located prior to the word to autocomplete.

Note that, in all cases, the word to autocomplete is not necessarily consecutive to these contexts.

Approaches to WLAC include modeling the task as a structured prediction (generation) task (Yang et al., 2022b; Ailem et al., 2022), modeling it as a segment-based interactive machine translation (IMT) task (Navarro et al., 2022), using pre-trained NMT models and available libraries (Moslem et al., 2022), and using a generator-reranker framework (Yang et al., 2022a).

In this work, we extended the segment-based IMT approach from Navarro et al. (2022) by adding a module based on a statistical dictionary that tackles zero-context completions—which are the cases in which, not having any feedback, the IMT system performs at its worst. Additionally, since this year edition allowed the use of pre-trained large language model (LLM), we experimented using the mT5 model (Xue et al., 2021).

2 Segment-based interactive machine translation

Segment-based IMT establishes a framework in which a human translator works together with the MT system to produce the final translation. This collaboration starts with the system proposing an initial translation hypothesis y_1^I of length I. Then, the user reviews this hypothesis and validates those sequence of words which they consider

Proceedings of the Eighth Conference on Machine Translation (WMT), pages 877–881
December 6–7, 2023.

to be correct ($\tilde{\mathbf{f}}_1, \ldots, \tilde{\mathbf{f}}_N$; where N is the number of non-overlapping validated segments). After that, they are able to merge two consecutive segments $\tilde{\mathbf{f}}_i$, $\tilde{\mathbf{f}}_{i+1}$ into a new one. Finally, they correct a word—which introduces a new one-word validated segment, $\tilde{\mathbf{f}}_i$, which is inserted in $\tilde{\mathbf{f}}_1^N$. This correction can also consist in a partially typed word $\tilde{\mathbf{f}}'_i$, in which case the system would complete it as part of its prediction.

The system's reacts to this user feedback by generating 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:

$$\begin{cases} \hat{y}_1^I = \tilde{\mathbf{f}}_1, \widehat{\mathbf{g}}_1, \ldots, \tilde{\mathbf{f}}'_i \widehat{\mathbf{g}}_i, \ldots, \tilde{\mathbf{f}}_N, \widehat{\mathbf{g}}_N \text{ if } \tilde{\mathbf{f}}'_i \in \tilde{\mathbf{f}}_1^N \\ \hat{y}_1^I = \tilde{\mathbf{f}}_1, \widehat{\mathbf{g}}_1, \ldots, \tilde{\mathbf{f}}_N, \widehat{\mathbf{g}}_N \text{ otherwise} \end{cases} \tag{1}$$

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

In this work, we extended Navarro et al. (2022)'s segment-based IMT approach by adding a new module that handles zero-context completions, which are harder for the IMT system to deal with (since there is no user feedback).

Additionally, we designed a new approach based on the mT5 LLM (Xue et al., 2021).

3.1 Segment-based IMT

Given a source sentence x_1^J, a sequence of typed characters $s_1^K = s_1, \ldots, s_K$ and a context $\mathbf{c} = \{\mathbf{c}_\text{l}, \mathbf{c}_\text{r}\}$, where $\mathbf{c}_\text{l} = c_{\text{l}1}, \ldots, c_{\text{l}S}$ and $\mathbf{c}_\text{r} = c_{\text{r}1}, \ldots, c_{\text{r}R}$; WLAC aims to autocomplete s_1^K to conform the word $w_1^W = s_1, \ldots, s_K, w_{K+1}, \ldots, w_W$. If we consider the context as the sequence of segments validated by the user ($\tilde{\mathbf{f}}_1^N = \mathbf{c}_\text{l}, \mathbf{c}_\text{r}$) and the sequence s_1^K as the partially-typed word correction (which would be inserted in $\tilde{\mathbf{f}}_1^N$ as a new one-word validated segment; leading to $\tilde{\mathbf{f}}_1^N = \mathbf{c}_\text{l}, s_1^K, \mathbf{c}_\text{r}$), we can view WLAC as a simplification of segment-based IMT. With that in mind, we can rewrite Eq. (1) as:

$$\hat{y}_1^I = \mathbf{c}_\text{l}, \widehat{\mathbf{g}}_1, s_1^K \widehat{\mathbf{g}}_2, \mathbf{c}_\text{r}, \widehat{\mathbf{g}}_3 \tag{4}$$

which, knowing that the prediction of the partially-typed correction corresponds to the first word of $\widehat{\mathbf{g}}_2$, can be rewritten as:

$$\hat{y}_1^I = \mathbf{c}_\text{l}, \widehat{\mathbf{g}}_1, s_1^K w_{K+1}^W, \widehat{\mathbf{g}}'_2, \mathbf{c}_\text{r}, \widehat{\mathbf{g}}_3 \tag{5}$$

Therefore, we can obtain the autocompleted word ($w_1^W = s_1^K w_{K+1}^W$) by performing a single step of the segment-based IMT protocol, discarding the rest of the translation prediction.

Zero-context

Since the core idea of IMT is reacting to a user feedback, not having any context results in the segment-based IMT approach performing at its worst. Thus, in this work we decided to create a special module dedicated to perform this kind of completion, using a variation of a statistical dictionary.

To that end, we computed *IBM's model 1* (Och and Ney, 2003) to obtain word alignments from source and target of the training set. Then, for each source word x_j, we compute the most probable translation t_a that starts with the sequence to complete s_1^K ($t_a = s_1, \ldots, s_K, t_{K+1}, \ldots, t_T$):

$$\hat{t}_j = \arg\max_{t_a} p(t_a | x_j) \tag{6}$$

where t_a belongs to the set of target words aligned with x_j that starts with s_1^K; and $p(t_a|x_j)$ is the alignment probability given by *IBM's model 1*.

Finally, we obtain the autocompleted word w_1^W as the most probable translation:

$$w_1^W = \arg\max_{t_1^J} p(t_1^J | x_1^J) \tag{7}$$

3.2 mT5

mT5 (Xue et al., 2021) is a multilingual variant of T5 (Raffel et al., 2020), pre-trained on a new Common Crawl-based dataset covering 101 languages. We choose to use this LLM since it has been pre-trained without any supervised training and, thus,

```
{
"src": "Indonesischer Lehrerin droht Haftstrafe wegen Dokumentation von sexueller Belästigung",
"context_type": "prefix",
"target": "school",
"typed_seq": "sch",
"left_context": "Indonesian",
"right_context": "",
"segment_id": "ref0"
},
```


(a) Original sentence in json format.

Indonesischer Lehrerin droht Haftstrafe wegen Dokumentation von sexueller Belästigung ||| Indonesian ||| ||| sch

(b) mT5 source sentence.

school

(c) mT5 target sentence.

Figure 1: Example of adapting a training sentence for fine-tuning mT5.

can be easily adapted to any downstream task by simply fine-tuning the model.

Therefore, this approach consists in fine-tuning mT5 for WLAC. To do so, we created a new parallel dataset in which source sentences are the concatenation of the original source sentence, the left context, the right context and the typed sequence (using a special token as a delimiter); and target sentences are the autocompletion. Fig. 1 shows an example.

4 Experimental setup

In this section, we present the details of our experimental session.

4.1 Evaluation

The WLAC 23 shared task selected accuracy as the automatic metric with which to report the evaluation of the different systems. This metric is computed as the total number of correctly predicted words normalized by the total number of words to complete:

$$\text{Acc} = N_{\text{match}}/N_{\text{all}} \tag{8}$$

where N_{match} is the number of predicted words that are identical to the human desired word, and N_{all} is the total number of testing words.

4.2 Corpora

We conducted our experiments using the English–German corpus provided by the organizers, which is a version of the WMT14's dataset, preprocessed by Stanford NLP Group.

Table 1: Statistics of the WLAC 2023 corpus. *Run.* stands for running, K for thousands and M for millions.

Partition	Characteristic	De	En
Training	Sentences	$4M$	
	Run. Words	$110M$	$116M$
	Vocabulary	$1.6M$	$800K$
Validation	Sentences	2000	
	Run. Words	$53K$	$53K$
	Vocabulary	$10.5K$	$7.5K$

For fine-tuning mT5 (see Section 3.2), we processed the training data using the provided script[1] in order to create the simulated data. We repeated this process multiple times to increase the number of samples. Table 2 presents the data statistics.

Table 2: Statistics of the synthetic corpus generated for fine-tuning the mT5 model. *Run.* stands for running, K for thousands and M for millions.

Partition	Characteristic	De	En
Training	Sentences	$50M$	
	Run. Words	$1627.6M$	$1677.6M$
	Vocabulary	$1.6M$	$800K$
Validation	Sentences	2000	
	Run. Words	$53K$	$53K$
	Vocabulary	$93.4K$	$144.9K$

[1] https://github.com/lemaoliu/WLAC/raw/main/scripts/generate_samples.py.

Table 3: Experimental results, measured in terms of accuracy.

Approach	Language	Overall	Prefix	Suffix	Bi-context	Zero-context
Segment-base IMT	De–En	0.400	0.453	0.151	0.395	0.570
	En-De	0.371	0.433	0.144	0.377	0.491
mT5	De–En	0.436	0.432	0.458	0.490	0.363
	En–De	0.374	0.373	0.389	0.431	0.301

4.3 Systems

The MT systems from our segment-based IMT approach were trained using *OpenNMT-py* (Klein et al., 2017). We selected a Transformer (Vaswani et al., 2017) architecture, with a word embedding size of 512. The hidden and output layers were set to 2048 and 512, respectively. Each multi-head attention layer has eight heads, and we stacked six encoder and decoder layers. We used Adam as the learning algorithm, with a learning rate of 2.0, b_1 of 0.9 and b_2 of 0.998. We set the batch size to 4096 tokens.

Additionally, we made use of the byte pair encoding (BPE) (Sennrich et al., 2016) algorithm, which was jointly trained on both languages of the dataset, applying a maximum number of 10.000 merges. Finally, we used our own implementation (based on *OpenNMT-py*) of segment-based IMT, which we adapted for WLAC. This implementation is openly available[2] for the benefit of the community.

For the mT5 approach, we made use of *HuggingFace's Transformer* (Wolf et al., 2019). Due to computing constrains, we selected *Google's mT5-base* model[3].

5 Results

Table 3 presents the official results of our approaches. We can see how both approaches yielded similar results. The main advantage of the segment-base IMT approach is that we can leverage an MT model for autocompletion by simply performing minor changes at the decoding step. However, looking at the results, while our zero-context proposal has successfully solved the problem of having no feedback, the system's performance significantly drops when the only available context is a suffix. In future works we shall address this behavior.

Regarding the mT5 approach, its main advantage is that we can adapt an already pre-trained mT5 model by simply performing fine-tuning with a WLAC dataset. With the exception of having no context, its behavior is constant for all kind of context. Additionally, it is worth remembering that we used *Google's mT5-base* model due to computing constrains. In a future work, we shall test how "bigger" mT5 models behave for this task.

6 Conclusions

In this work, we have presented our submission to WLAC shared task from WMT23. Our first proposal extended Navarro et al. (2022)'s segment-based IMT approach by adding a zero-context—based on a statistical dictionary—that handles separately the cases in which no context is given. This approach yielded satisfactory results for all cases except when the given context consists in a suffix.

Our second proposal consisted in leverage the pre-trained LLM model mT5 by performing a simple fine-tuning that enables the model to be used for WLAC, achieving satisfactory results for all type of contexts.

As a future work, we would like to study the behavior of the segment-based IMT approach when dealing with suffixes. Additionally we would like to consider the use of other LLM, as well as different versions of the mT5 model.

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

Melissa Ailem, Jingshu Liu, Jean-Gabriel Barthélemy, and Raheel Qader. 2022. Lingua custodia's

[2] https://github.com/PRHLT/OpenNMT-py/tree/word-level_autocompletion.

[3] https://huggingface.co/google/mt5-base.

participation at the WMT 2022 word-level autocompletion shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 1170–1175.

Francisco Casacuberta, George Foster, Guoping Huang, Philipp Koehn, Geza Kovacs, Lemao Liu, Shuming Shi, Taro Watanabe, and Chengqing Zong. 2022. Findings of the word-level autocompletion shared task in wmt 2022. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 812–820.

G. Klein, 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.

Huayang Lin, Lemao Liu, Guoping Huang, and Shuming Shi. 2021. GWLAN: General word-level autocompletion for computer-aided translation. In *Proceedings of the Joint Conference of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing*. In Press.

Arle Lommel. 2018. Augmented translation: A new approach to combining human and machine capabilities. In *Proceedings of the Conference of the Association for Machine Translation in the Americas. Volume 2: User Track*, pages 5–12.

Yasmin Moslem, Rejwanul Haque, and Andy Way. 2022. Translation word-level auto-completion: What can we achieve out of the box? In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 1176–1181.

Ángel Navarro, Miguel Domingo, and Francisco Casacuberta. 2022. PRHLT's submission to WLAC 2022. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 1182–1186.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational linguistics*, 29(1):19–51.

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 1715–1725.

Antonio Toral. 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.

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 *Advances in Neural Information Processing Systems*, pages 5998–6008.

Heidi Verplaetse and An Lambrechts. 2019. Surveying the use of CAT tools, terminology management systems and corpora among professional translators: general state of the art and adoption of corpus support by translator profile. *Parallèles*, 31(2):3–31.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*.

Cheng Yang, Siheng Li, Chufan Shi, and Yujiu Yang. 2022a. Iigroup submissions for wmt22 word-level autocompletion task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 1187–1191.

Hao Yang, Hengchao Shang, Zongyao Li, Daimeng Wei, Xianghui He, Xiaoyu Chen, Zhengzhe Yu, Jiaxin Guo, Jinlong Yang, Shaojun Li, et al. 2022b. Hw-tsc's submissions to the wmt22 word-level auto completion task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 1192–1197.

Jost Zetzche. 2007. Translation memory: state of the technology. *Multilingual*, 18:34–38.