Findings of the Word-Level AutoCompletion Shared Task in WMT 2022*

Francisco Casacuberta [1] **George Foster** [2] **Guoping Huang** [3] **Philipp Koehn** [4,5]
Geza Kovacs [6] **Lemao Liu** [3] **Shuming Shi** [3] **Taro Watanabe** [7] **Chengqing Zong** [8]
[1] Universitat Politècnica de València [2] Google [3] Tencent AI Lab [4] Johns Hopkins University
[5] Meta AI [6] LILT [7] Nara Institute of Science and Technology
[8] Institute of Automation, Chinese Academy of Sciences

Abstract

Recent years have witnessed rapid advancements in machine translation, but the state-of-the-art machine translation system still can not satisfy the high requirements in some rigorous translation scenarios. Computer-aided translation (CAT) provides a promising solution to yield a high-quality translation with a guarantee. Unfortunately, due to the lack of popular benchmarks, the research on CAT is not well developed compared with machine translation. In this year, we hold a new shared task called Word-level AutoCompletion (WLAC) for CAT in WMT. Specifically, we introduce some resources to train a WLAC model, and particularly we collect data from CAT systems as a part of test data for this shared task. In addition, we employ both automatic and human evaluations to measure the performance of the submitted systems, and our final evaluation results reveal some findings for the WLAC task.

1 Introduction

In past decades, the machine translation community has witnessed a significant evolution from statistical machine translation (Koehn et al., 2003; Chiang, 2005; Koehn, 2009b) to neural machine translation (NMT) (Sutskever et al., 2014; Bahdanau et al., 2015; Wu et al., 2016; Gehring et al., 2017; Vaswani et al., 2017). NMT has achieved a rapid and tremendous advancement in translation performance (Barrault et al., 2019). Despite its success in many real-world applications, its translation quality still can not satisfy the high requirements in some scenarios. In such rigorous scenarios, one promising approach is to leverage machines to assist human translation, such as Computer-aided Translation (CAT) (Bowker, 2002; Koehn, 2009a; Foster et al., 1997; Langlais et al., 2000; Barrachina et al., 2009; Alabau et al., 2014; Knowles and Koehn, 2016; Santy et al., 2019).

However, the development in CAT is much slower than in machine translation. For example, there are hundreds of research papers on machine translation in natural language processing conferences each year, whereas only a few papers on CAT are published. One of the main reasons is that there are few popular benchmarks or shared tasks for CAT research, which enable researchers to make continuous progress in this area. Consequently, in WMT this year, we hold a new shared task, Word-level AutoCompletion (WLAC), to facilitate the research in CAT. Generally, WLAC aims to autocomplete a word when a human translator types a sequence of characters (Huang et al., 2015; Li et al., 2021). As a basic functionality in many CAT systems, WLAC is used to accelerate the editing process for human translators and it plays an important role in CAT.

In this paper, we describe the overview for the shared task of WLAC in WMT 2022, such as task description, datasets, participants and their evaluations. The shared task involves two language pairs, including Chinese-English and German-English and it contains four subtasks corresponding to all four directional pairs. For data preparation, since it is too costly to collect realistic data with a considerable scale to train WLAC models, we follow the standard practice to construct the training data from a bilingual corpus by simulation. Moreover, to make the testing stage resemble the realistic scenario in CAT, we collect some data from two CAT systems as a part of test data. In this shared task, we receive 27 submissions in total for all subtasks from five participants which are quickly summarized in this paper. To evaluate the submissions, we particularly conduct human evaluation in addition to automatic evaluation. After evaluation, we finally obtain some findings from the submission results, which we hope may inspire future advancements for the WLAC task.

* The authors are listed alphabetically.

Proceedings of the Seventh Conference on Machine Translation (WMT), pages 812 - 820

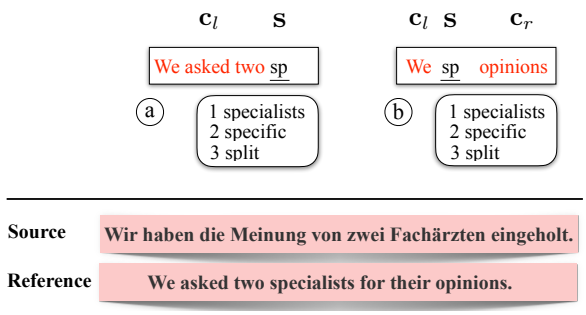


Figure 1: Illustration of WLAC task. The translation context $\boldsymbol{c}$ for a source sentence $\boldsymbol{x}$ includes the left context $\boldsymbol{c}_l$ and right context $\boldsymbol{c}_r$, underlined text "sp" is the human typed characters $\boldsymbol{s}$ and the words in the rounded rectangles are word-level autocompletion candidates.

2 Task Description and Data Preparation

2.1 Task Description

Suppose $\boldsymbol{x}$ is a source sequence, $\boldsymbol{s}$ is a sequence of human typed characters and $\boldsymbol{c} = (\boldsymbol{c}_l, \boldsymbol{c}_r)$ is a translation context. The translation pieces $\boldsymbol{c}_l$ and $\boldsymbol{c}_r$ are on the left and right hand sides of $\boldsymbol{s}$, respectively. Formally, given the tuple $(\boldsymbol{x}, \boldsymbol{c}, \boldsymbol{s})$, the WLAC task aims to predict the target word w with $\boldsymbol{s}$ as its prefix, which is the most appropriate to be placed between $\boldsymbol{c}_l$ and $\boldsymbol{c}_r$ (Huang et al., 2015; Li et al., 2021).

To make the task more general in real-world scenarios, WLAC task assumes that the left context $\boldsymbol{c}_l$ and right context $\boldsymbol{c}_r$ can be empty, which leads to the following four types of context:

- Zero-context: both $\boldsymbol{c}_l$ and $\boldsymbol{c}_r$ are empty;
- Suffix: $\boldsymbol{c}_l$ is empty;
- Prefix: $\boldsymbol{c}_r$ is empty;
- Bi-context: neither $\boldsymbol{c}_l$ nor $\boldsymbol{c}_r$ is empty.

Figure 1 ⓐ and ⓑ show two examples about the WLAC task. According to the above criterion, Figure 1 ⓐ belongs to Prefix type and Figure 1 ⓑ belongs to Bi-context type.

2.2 Data Preparation

The WLAC task in WMT2022 involves following two language pairs: English⇔Chinese and English⇔German. Each language pair corresponds to two directional subtasks, leading to four subtasks.

Training Data In fact, it is too costly to manually annotate the training dataset consisting of tuples $\langle \boldsymbol{x}, \boldsymbol{c}, \boldsymbol{s}, w \rangle$ for WALC task. We alternatively follow Li et al. (2021) to construct the simulated training data for WLAC from existing bilingual data.[1] The key idea of such simulation is that it randomly samples a target word w and context $\boldsymbol{c}$ from the reference translation $\boldsymbol{y}$ of $\boldsymbol{x}$, a human typed sequence $\boldsymbol{c}$ for the target word w to obtain an example, e.g., a tuple of $\langle \boldsymbol{x}, \boldsymbol{c}, \boldsymbol{s}, w \rangle$.

	En-De	En-Zh
Sentences	4,465,840	15,886,041
Words	120M/114M	441M/395M

Table 1: The statistics of English-German and English-Chinese bilingual datasets for training.

For training on English-German language pair, we use the WMT14 En-De training dataset preprocessed by Stanford NLP Group[2], which consists of about 4.5M sentence pairs. For training on English-Chinese language pair, we take the "UN Parallel Corpus V1.0" dataset[3] from WMT17 consisting of 15M sentence pairs. We use Moses scripts[4] to tokenize English and German sentences and jieba[5] to segment Chinese words for each sentence. The detailed statistics of bilingual datasets are shown in Table 1.

Note that in this shared task, participants must use the above bilingual data and it is illegal to any other bilingual data beyond. However, to achieve better performance, any monolingual data is allowed as well as the pre-trained language models such as BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019).

Test Data To ensure authenticity and reliability, the test data for WLAC is not from existing open-source bilingual data. We create the test data by ourselves: the test datasets are obtained in two different ways, leading to two types of test data. One type (**Type I**) is the simulation on bilingual data similar to the creation of training data and the other type (**Type II**) is from CAT translation systems.

For the Type I test data, we first create a new bilingual dataset and then obtain the simulated tuples $\langle \boldsymbol{x}, \boldsymbol{c}, \boldsymbol{s}, w \rangle$ from the bilingual dataset. Specifically, to ensure that the ground-truth word w is not

[1]The scripts for simulation is available at https://github.com/lemaoliu/WLAC.

[2]https://nlp.stanford.edu/projects/nmt/data

[3]https://conferences.unite.un.org/UNCorpus/Home/DownloadOverview

[4]https://github.com/moses-smt/mosesdecoder

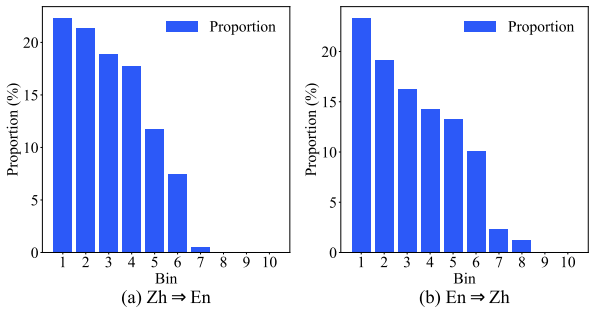
[5]https://github.com/fxsjy/jieba

	Zh⇒En	En⇒Zh	De⇒En	En⇒De
		Sentences		
Type I	5434	6122	5700	-
Type II	2109	1953	1996	13418
Overall	**7543**	**8075**	**7696**	**13418**
		Words		
Type I	615K/115K	662K/109K	519K/96K	-
Type II	242K/45K	237K/38K	203K/38K	437K/85K
Overall	**857K/161K**	**899K/147K**	**722K/134K**	**437K/85K**

Table 2: The statistics (number of sentences and words) of Zh⇒En, En⇒Zh, De⇒En and En⇒De test datasets. A/B denotes that A is the total number of source words in the source sentences and B is the total number of target words in the context.

	Zh⇒En	En⇒Zh	De⇒En	En⇒De
		Bi-context		
Type I	5102	5137	5313	-
Type II	2092	1676	1950	6514
Overall	**7194**	**6813**	**7263**	**6514**
		Prefix		
Type I	5330	5249	5686	-
Type II	2087	1645	1968	6319
Overall	**7417**	**6894**	**7654**	**6319**
		Suffix		
Type I	5053	5156	5382	-
Type II	2089	1674	1994	6571
Overall	**7142**	**6830**	**7376**	**6571**
		Zero-context		
Type I	5200	5137	5256	-
Type II	2098	1622	2047	6491
Overall	**7298**	**6759**	**7303**	**6491**

Table 3: The statistics (number of $\langle \boldsymbol{x}, \boldsymbol{c}, \boldsymbol{s}, w \rangle$ tuples) of different context types on WLAC test datasets (including both Type I and Type II parts).



(a) The proportion on Zh⇒En **(b)** The proportion on En⇒Zh

Figure 2: The proportion of the bins of w typed by human translators from CAT systems according to word frequency in bilingual corpus on German-English language pair. Bin 1 and Bin 10 respectively denote the most infrequent word bin and the most frequent bin.

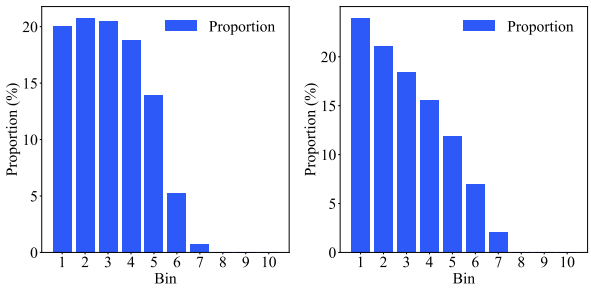
exposed to the training data, we first crawl bilingual news from Internet websites in the latest 3 years. After crawling the raw bilingual data, we employ professional translators to check and screen the low quality bilingual data to obtain high-quality bilingual sentences. Finally, we follow the simulation way to obtain the training tuples $\langle \boldsymbol{x}, \boldsymbol{c}, \boldsymbol{s}, w \rangle$ based on the crawled bilingual data described above.

The Type II test data is collected from two CAT systems LILT[6] and TranSmart[7] (Huang et al., 2021). Specifically, given a source sentence $\boldsymbol{x}$, a human translator works on a CAT system to generate a translation $\boldsymbol{y}$. In the log file from the CAT system, only the information about $w \in \boldsymbol{y}$ typed by human is stored, while other dynamic information such as typed characters and context for each w is not available. Therefore, we create both $\boldsymbol{c}$ and $\boldsymbol{s}$ from $\boldsymbol{y}$ for each w by simulation as before. In other words, for each example $\langle \boldsymbol{x}, \boldsymbol{c}, \boldsymbol{s}, w \rangle$, both $\boldsymbol{c}$ and $\boldsymbol{s}$ are simulated but w is realistic. Note that each sentence from the Type II data is also not included in the training data.

For En⇒De task, the entire test data is the type II from the CAT system LILT. For Zh⇒En, En⇒Zh and De⇒En tasks, the test data is the combination of both types, i.e., some test data is Type I from the simulation over bilingual data and the other test data is Type II from the CAT system TranSmart.

To pre-process the test data (e.g., word tokenization), we adopt the same pre-processing way as used in training data. Table 2 summarizes the detailed statistics in terms of sentences and words for the test data, and Table 3 reports the number of ex-

[6] https://lilt.com
[7] https://transmart.qq.com



(a) De ⇒ En (b) En ⇒ De

(a) The proportion on De⇒En **(b)** The proportion on En⇒De

Figure 3: The proportion of the bins of w typed by human translators from CAT systems according to word frequency in bilingual corpus on German-English language pair. Bin 1 and Bin 10 respectively denote the most infrequent word bin and the most frequent bin.

amples for four different context types in test data. Note that each source sentence $\boldsymbol{x}$ may correspond to multiple examples $\langle \boldsymbol{x}, \boldsymbol{c}, \boldsymbol{s}, w \rangle$ and thus, the total number of sentences in Table 2 is not the same as the total number of examples in Table 3.

Furthermore, one may be curious about the characteristics of the words typed by human translators. We understand the human typed words from the perspective of word frequency. We first group the target vocabulary into ten bins with equal size according to word frequency computed in the bilingual corpus, we collect all typed words w together and then assign a bin for each word, and finally we calculate the proportion of each bin. The statistics are shown in Table 2 and Table 3, where bin 1 denotes the most infrequent words while bin 10 denotes the most frequent words. From these tables, it is observed that human translators usually type infrequent words. This observation is reasonable because it is easy for machine translation systems to make a correct translation decision on a frequent word.

3 Evaluation Metric

We use both automatic evaluation and human evaluation to measure all submitted systems.

Automatic Evaluation To measure the performance of the submitted systems, we choose accuracy as the automatic evaluation metric (Li et al., 2021) as follows :

$$\text{ACC} = \frac{N_{match}}{N_{all}} \quad (1)$$

where N_{match} is the number of correct predicted words and N_{all} is the number of **all** test examples.

Although automatic evaluation is convenient, it still has some limitations because there may be multiple ground-truth words w (i.e., ground truth is a word set) which suffice to the constraint of $\boldsymbol{s}$ and are compatible with $\langle \boldsymbol{x}, \boldsymbol{c} \rangle$, especially for a short $\boldsymbol{c}$ and $\boldsymbol{s}$. For instance, when $\boldsymbol{c}$ and $\boldsymbol{s}$ are empty, any translation of a source word in $\boldsymbol{x}$ may be a ground-truth word if it suffices to the constraint of $\boldsymbol{s}$. Therefore, we additionally conduct human evaluation for more faithful evaluation on the submissions.

Human Evaluation Human evaluation is appealing, but it is too costly to evaluate all testing examples. Instead, we conduct human evaluation on a small subset of test data for efficiency. Specifically, for all four subtasks, we randomly sample **400** test examples derived from the Type II part of the test data as the human evaluation dataset. After participants submit their systems, we gather their predicted words to constitute a prediction set for each test example. Then we hire professional translators to annotate the correct ones in the prediction set. Finally, we use the manually annotated ground-truth word set to re-evaluate submitted systems and the human score is defined by the percentage of predicted words annotated as correct words by human. Since more than one target word can be annotated as the correct word, the human evaluation score is higher than the automatic score in general.

4 Submitted Systems and Results

In this year, there are five teams participating in this shared task and we receive 27 submissions from them. In this section, first, we quickly describe the participants and their submitted systems, then we present their evaluation results in terms of both automatic and human evaluations, and finally, we shed light on some findings according to evaluation results.

4.1 Participants and Submitted Systems

HW-TSC (Yang et al., 2022b) The Huawei Translation Services Center (HW-TSC) participates in Zh⇒En, De⇒En and En⇒De language directions. They model the WLAC task as a structured prediction (or generation) task, which iteratively generates a subword to compose the prediction word. Specifically, they first train a vanilla Transformer on machine translation task as a baseline. Then they fine-tune the baseline with WLAC data and BERT-style MLM data to get the final model.

Systems	Fullset	Subset	
	Acc. (Rank)	Human. (Rank)	Acc. (Rank)
HW-TSC	59.40 (#1)	91.25 (#1)	69.25 (#1)
THU IIGroup-1	54.05 (#2)	85.00 (#6)	59.75 (#6)
THU IIGroup-2	51.11 (#3)	83.75 (#7)	57.25 (#7)
DCU-NCI-4	50.41 (#4)	86.75 (#3)	63.25 (#2)
DCU-NCI-3	50.26 (#5)	86.75 (#3)	62.25 (#3)
DCU-NCI-2	49.35 (#6)	86.00 (#5)	61.75 (#5)
DCU-NCI-1	49.06 (#7)	87.00 (#2)	62.00 (#4)

Table 4: Official results of WLAC task for Zh⇒En. Acc. and Human. represent automatic and human evaluations, respectively. Rank denotes the ranking according to the corresponding metric.

Systems	Fullset	Subset	
	Acc. (Rank)	Human. (Rank)	Acc. (Rank)
THU IIGroup-1	53.98 (#1)	83.25 (#1)	54.50 (#1)
THU IIGroup-2	48.90 (#2)	77.50 (#2)	48.75 (#2)
DCU-NCI-2	31.94 (#3)	57.75 (#3)	37.75 (#4)
DCU-NCI-1	31.94 (#4)	57.25 (#4)	38.00 (#3)

Table 5: Official results of WLAC task for En⇒Zh. Acc. and Human. represent automatic and human evaluations, respectively. Rank denotes the ranking according to the corresponding metric.

It is worth noting that they use a character embedding method to encode the information of a human typed sequence to the model. Moreover, they adopt some basic strategies to improve the performance, including back translation, averaging and ensemble techniques.

PRHLT (Ángel Navarro et al., 2022) The team of PRHLT submits their systems for De⇒En and En⇒De subtasks. They first cast the WLAC task as a segment-based IMT task. More concretely, they consider the translation context as the sequence of segments validated by the user in IMT and the sequence of human typed characters as partially-typed word correction. They experiment with both RNN architecture and Transformer architecture.

DCU-NCI (Moslem et al., 2022) DCU-NCI proposes to address the WLAC task with the help of pre-trained NMT models and available libraries, which is a new way to solve the WLAC task. Their systems do not need any additional training to address the WLAC task.Specifically, they use OPUS pre-trained models[8] and employ CTranslate2 [9] as an inference engine. During the decoding stage, they find that random sampling restricted with the best 10 candidates perform better than beam search. Furthermore, they also try to adopt different sampling temperatures (ST) to change the randomness of the generation. We denote the system trained with ST=1.0 as DCU-NCI-1, the system with ST=1.3 as DCU-NCI-2, the system with ST=1.3 and detokenization as DCU-NCI-3 and the system trained with ST=1.0 and detokenization as DCU-NCI-4.

Lingua Custodia (Ailem et al., 2022) The team of Lingua Custodia submits systems for De⇒En and En⇒De tracks. They also treat the WLAC task as a structured prediction task and adopt the Transformer architecture for generation. Specifically, they use a Transformer Encoder to encode the source sentence, translation context and human typed characters, and a Transformer Decoder to generate a sequence of subwords to constitute a target word step by step. In addition, they propose several data-cleaning strategies to pre-process the bilingual translation data. We denote the system trained with the initial corpus as Lingua Custodia-1 and the system trained with the cleaned corpus as Lingua Custodia-2.

[8] https://github.com/Helsinki-NLP/Tatoeba-Challenge

[9] https://github.com/OpenNMT/CTranslate2

Systems	Fullset	Subset	
	Acc. (Rank)	Human. (Rank)	Acc. (Rank)
HW-TSC	62.06 (#1)	87.50 (#3)	78.00 (#3)
DCU-NCI-1	61.44 (#2)	88.50 (#2)	80.50 (#1)
DCU-NCI-2	60.92 (#3)	88.75 (#1)	79.00 (#2)
Lingua Custodia-1	57.36 (#4)	76.75 (#5)	67.50 (#5)
THU IIGroup-1	57.27 (#5)	78.75 (#4)	69.75 (#4)
Lingua Custodia-2	54.85 (#6)	74.50 (#7)	63.50 (#7)
THU IIGroup-2	54.32 (#7)	76.25 (#6)	66.50 (#6)
PRHLT	39.02 (#8)	51.25 (#8)	44.25 (#8)

Table 6: Official results of WLAC task for De⇒En. Acc. and Human. represent automatic and human evaluations, respectively. Rank denotes the ranking according to the corresponding metric.

Systems	Fullset	Subset	
	Acc. (Rank)	Human. (Rank)	Acc. (Rank)
HW-TSC	63.82 (#1)	79.00 (#1)	66.75 (#1)
DCU-NCI-1	58.94 (#2)	67.25 (#2)	56.00 (#2)
DCU-NCI-2	58.49 (#3)	65.50 (#3)	56.75 (#3)
Lingua Custodia-1	48.97 (#4)	61.75 (#4)	52.25 (#4)
Lingua Custodia-2	48.44 (#5)	61.00 (#5)	50.75 (#5)
THU IIGroup-1	41.83 (#6)	55.50 (#6)	46.00 (#6)
THU IIGroup-2	40.69 (#7)	53.50 (#7)	44.75 (#7)
PRHLT	33.97 (#8)	45.75 (#8)	37.00 (#8)

Table 7: Official results of WLAC task for En⇒De. Acc. and Human. represent automatic and human evaluations, respectively. Rank denotes the ranking according to the corresponding metric.

THU IIGroup (Yang et al., 2022a) THU IIGroup participates in Zh⇒En, En⇒Zh, De⇒En and En⇒De directions. They propose a generator-reranker framework to tackle the WLAC task. Specifically, they adopt the baseline model based on Transformer as a generator to yield a set of candidate words. Moreover, they additionally train a reranking model to rerank the candidate words to get the final prediction. We denote the generator as THU IIGroup-1 and the reranker as THU IIGroup-2.

Summary on submitted systems All submitted systems in this year choose the powerful Transformer architecture by stacking multiple layers of attention as the backbone for the WLAC task. To tackle the constraint of the human typed character sequence, some submitted systems consider it as a hard constraint while others (HW-TSC and Lingua Custodia) considering it as a soft constraint: they differ in that the model architecture in the former is aware of the constraints but the later matters. In addition, most systems formalize the WLAC task as a classification task where the target word w is actually a label, but one system (HW-TSC) treats WLAC as a structured prediction task: the target w is decomposed into a sequence of BPE units and it is beneficial to predict the out-of-vocabulary words.

4.2 Evaluation Results

Since human evaluation is only conducted on the partial test dataset consisting of 400 examples and automatic evaluation can be evaluated on both the full and partial test datasets, we evaluate all the submitted systems on two different types of test data, i.e., full test data set and partial test data set as follows. All of their results on Zh⇒En, En⇒Zh, De⇒En and En⇒De are listed in Table 4,5,6 and 7.

Results on Full Test Set From the four tables, it can be shown that the systems of HW-TSC shows impressive performance and achieve the best for Zh⇒En, De⇒En and En⇒De, and THU IIGroup

yields the best performance for En⇒Zh. As we can see, there are some gaps in performance among different systems, which means there is a significant opportunity for growth in the WLAC task.

Results on Partial Test Set As described in Section 3, it is not surprising that human evaluation scores are much higher than automatic evaluation scores. In addition, it is observed that on the partial test set, the human evaluation results are almost in line with the automatic evaluation result although there indeed is a slight inconsistency. This fact demonstrates that automatic evaluation metric can act as a good alternative for evaluation. Moreover, it is interesting that, in terms of automatic evaluation, the rankings between the full and partial test datasets are clearly different on Zh⇒En, although they are mostly consistent on other tasks. This observation indicates that a small test dataset may lead to a biased conclusion.

4.3 Discussion

In this section, we shed light on some key findings among all the submitted systems which we hope will push forward the development of the WLAC task in the future.

First, it would be preferable to treat the WLAC task as a structured prediction task rather than a classification task according to the prediction accuracy. One advantage of the structured prediction perspective is that it can decompose the predicted word into a sequence of tokens at the subword level to tackle out-of-vocabulary words. This is appealing specially because most of the typed words by human translators are low frequent words as observed in our analysis. However, it is noteworthy that a structured predition model requires more computing time than a classification model during the inference stage.

Second, WLAC task may benefit from NMT based pre-training. It is noticed that one participant employs such a pre-training strategy: it first trains a standard NMT model on the bilingual dataset and then it fine-tunes the model with the WLAC data to obtain a WLAC model. It is reasonable since in NMT task, every token in the target-side serves as a label, while in WLAC task, only the target token serves as a label. The former can facilitate the training procedure and provide a good weight initialization for WLAC tailored model.

Third, leveraging monolingual data is a common practice to improve the performance in many NLP tasks, including machine translation. For example, a pre-trained model trained on monolingual data such as XLM (Lample and Conneau, 2019) and MASS (Song et al., 2019) are successful to improve translation quality, and back translation (Sennrich et al., 2015; Edunov et al., 2018) is also an effective strategy by construction synthetic bilingual data from target monolingual data. In WLAC task, one participant tries to enhance the WLAC model by using back translation similar to NMT and it is promising to design new ways customized for WLAC.

5 Conclusion

Word-level AutoCompletion is a basic functionality in computer-aided translation systems to facilitate the editing efficiency for translators. In WMT this year, the Word-level AutoCompletion shared task is introduced and it covers two language pairs including four directional subtasks. We provide high-quality test datasets and human evaluation to evaluate different systems fairly. On all subtasks we receive 27 submissions from five participants which address the WLAC task from different perspectives. Automatic and human evaluations on these submissions reveal some key findings which may provide valuable insights for future research on this task. Finally, we hope that WLAC task will attract more researchers to participate in the exploration of computer-aided translation.

Acknowledgements

We would like to thank Huayang Li for valuable discussions and Cheng Yang for data pre-processing on the shared task, and especially appreciate the annotators for their creating test data and manual evaluation on the submitted systems. In addition, we thank the participants for their contributions on this shared task.

References

Melissa Ailem, Jinghsu Liu, Jean-Gabriel Barthélemy, and Raheel Qader. 2022. Lingua custodia's participation at the wmt 2022 word-level auto-completion shared task. In *Proceedings of the Seventh Conference on Machine Translation Shared Tasks Papers*. Association for Computational Linguistics.

Vicent Alabau, Christian Buck, Michael Carl, Francisco Casacuberta, Mercedes García-Martínez, Ulrich Germann, Jesús González-Rubio, Robin Hill, Philipp Koehn, Luis A Leiva, et al. 2014. Casmacat: A

computer-assisted translation workbench. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Sergio Barrachina, Oliver Bender, Francisco Casacuberta, Jorge Civera, Elsa Cubel, Shahram Khadivi, Antonio L. Lagarda, Hermann Ney, Jesús Tomás, Enrique Vidal, and Juan Miguel Vilar. 2009. Statistical approaches to computer-assisted translation. *Comput. Linguistics*, 35(1):3–28.

Loïc Barrault, Ondřej Bojar, Marta R Costa-Jussa, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, et al. 2019. Findings of the 2019 conference on machine translation (wmt19). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 1–61.

Lynne Bowker. 2002. *Computer-aided translation technology: A practical introduction*. University of Ottawa Press.

David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 263–270.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 489–500. Association for Computational Linguistics.

George Foster, Pierre Isabelle, and Pierre Plamondon. 1997. Target-text mediated interactive machine translation. *Machine Translation*, 12(1):175–194.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1243–1252.

Guoping Huang, Lemao Liu, Xing Wang, Longyue Wang, Huayang Li, Zhaopeng Tu, Chengyan Huang, and Shuming Shi. 2021. Transmart: A practical interactive machine translation system. *CoRR*, abs/2105.13072.

Guoping Huang, Jiajun Zhang, Yu Zhou, and Chengqing Zong. 2015. A new input method for human translators: integrating machine translation effectively and imperceptibly. In *IJCAI*.

Rebecca Knowles and Philipp Koehn. 2016. Neural interactive translation prediction. In *12th Conferences of the Association for Machine Translation in the Americas: MT Researchers' Track, AMTA 2016, Austin, TX, USA, October 28 - November 1, 2016*, pages 107–120. The Association for Machine Translation in the Americas.

Philipp Koehn. 2009a. A process study of computer-aided translation. *Machine Translation*, 23(4):241–263.

Philipp Koehn. 2009b. *Statistical machine translation*. Cambridge University Press.

Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 127–133.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*.

Philippe Langlais, George Foster, and Guy Lapalme. 2000. Transtype: a computer-aided translation typing system. In *ANLP-NAACL 2000 Workshop: Embedded Machine Translation Systems*.

Huayang Li, Lemao Liu, Guoping Huang, and Shuming Shi. 2021. GWLAN: general word-level autocompletion for computer-aided translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 4792–4802. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Yasmin Moslem, Rejwanul Haque, and Andy Way. 2022. Word-level auto-completion: What can we achieve out of the box? In *Proceedings of the Seventh Conference on Machine Translation Shared Tasks Papers*. Association for Computational Linguistics.

Ángel Navarro, Miguel Domingo, and Francisco Casacuberta. 2022. Prhlt's submission to wlac 2022. In *Proceedings of the Seventh Conference on Machine Translation Shared Tasks Papers*. Association for Computational Linguistics.

Sebastin Santy, Sandipan Dandapat, Monojit Choudhury, and Kalika Bali. 2019. INMT: interactive neural machine translation prediction. In *Proceedings*

of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019 - System Demonstrations, pages 103–108. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. Mass: Masked sequence to sequence pre-training for language generation. *arXiv preprint arXiv:1905.02450*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 3104–3112.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

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 Shared Tasks Papers*. Association for Computational Linguistics.

Hao Yang, Hengchao Shang, Zongyao Li, Daimeng Wei, Xianghui He, Xiaoyu Chen, Zhengzhe Yu, Jiaxin Guo, Jinlong Yang, Shaojun Li, Yuanchang Luo, Yuhao Xie, Lizhi Lei, and Ying Qin. 2022b. Hw-tsc's submissions to the wmt22 word-level auto completion task. In *Proceedings of the Seventh Conference on Machine Translation Shared Tasks Papers*. Association for Computational Linguistics.