CUNI-KIT System for Simultaneous Speech Translation Task at IWSLT 2022

Peter Polák[1] and **Ngoc-Quan Ngoc**[2] and **Tuan-Nam Nguyen**[2] and **Danni Liu**[3]
Carlos Mullov[2] and **Jan Niehues**[2] and **Ondřej Bojar**[1] and **Alexander Waibel**[2,4]
polak@ufal.mff.cuni.cz
[1] Charles University
[2] Karlsruhe Institute of Technology
[3] Maastricht University
[4] Carnegie Mellon University

Abstract

In this paper, we describe our submission to the Simultaneous Speech Translation at IWSLT 2022. We explore strategies to utilize an offline model in a simultaneous setting without the need to modify the original model. In our experiments, we show that our onlinization algorithm is almost on par with the offline setting while being 3× faster than offline in terms of latency on the test set. We also show that the onlinized offline model outperforms the best IWSLT2021 simultaneous system in medium and high latency regimes and is almost on par in the low latency regime. We make our system publicly available.[1]

1 Introduction

This paper describes the CUNI-KIT submission to the Simultaneous Speech Translation task at IWSLT 2022 (Anastasopoulos et al., 2022) by Charles University (CUNI) and Karlsruhe Institute of Technology (KIT).

Recent work on end-to-end (E2E) simultaneous speech-to-text translation (ST) is focused on training specialized models specifically for this task. The disadvantage is the need of storing an extra model, usually a more difficult training and inference setup, increased computational complexity (Han et al., 2020; Liu et al., 2021) and risk of performance degradation if used in offline setting (Liu et al., 2020a).

In this work, we base our system on a robust multilingual offline ST model that leverages pretrained wav2vec 2.0 (Baevski et al., 2020) and mBART (Liu et al., 2020b). We revise the onlinization approach by Liu et al. (2020a) and propose an improved technique with a fully controllable quality-latency trade-off. We demonstrate that without any change to the offline model, our simultaneous system in the mid- and high-latency regimes is on par with the offline performance. At the same time, the model outperforms previous IWSLT systems in medium and high latency regimes and is almost on par in the low latency regime. Finally, we observe a problematic behavior of the average lagging metric for speech translation (Ma et al., 2020) when dealing with long hypotheses, resulting in negative values. We propose a minor change to the metric formula to prevent this behavior.

Our contribution is as follows:

- We revise and generalize onlinization proposed by Liu et al. (2020a); Nguyen et al. (2021) and discover parameter enabling quality-latency trade-off,
- We demonstrate that one multilingual offline model can serve as simultaneous ST for three language pairs,
- We demonstrate that an improvement in the offline model leads also to an improvement in the online regime,
- We propose a change to the average lagging metric that avoids negative values.

2 Related Work

Simultaneous speech translation can be implemented either as a (hybrid) cascaded system (Kolss et al., 2008; Niehues et al., 2016; Elbayad et al., 2020; Liu et al., 2020a; Bahar et al., 2021) or an end-to-end model (Han et al., 2020; Liu et al., 2021). Unlike for the offline speech translation where cascade seems to have the best quality, the end-to-end speech translation offers a better quality-latency trade-off (Ansari et al., 2020; Liu et al., 2021; Anastasopoulos et al., 2021).

End-to-end systems use different techniques to perform simultaneous speech translation. Han et al. (2020) uses wait-k (Ma et al., 2019) model and metalearning (Indurthi et al., 2020) to alleviate

[1] https://hub.docker.com/repository/docker/polape7/cuni-kit-simultaneous

Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022), pages 277 - 285

the data scarcity. Liu et al. (2020a) uses a unidirectional encoder with monotonic cross-attention to limit the dependence on future context. Other work (Liu et al., 2021) proposes Cross Attention augmented Transducer (CAAT) as an extension of RNN-T (Graves, 2012).

Nguyen et al. (2021) proposed a hypothesis stability detection for automatic speech recognition (ASR). The *shared prefix* strategy finds the longest common prefix in all beams. Liu et al. (2020a) explore such strategies in the context of speech recognition and translation. The most promising is the longest common prefix of two consecutive chunks. The downside of this approach is the inability to parametrize the quality-latency trade-off. We directly address this in our work.

3 Onlinization

In this section, we describe the onlinization of the offline model and propose two ways to control the quality-latency trade-off.

3.1 Incremental Decoding

Depending on the language pair, translation tasks may require reordering or a piece of information that might not be apparent until the source utterance ends. In the offline setting, the model processes the whole utterance at once, rendering the strategy most optimal in terms of quality. If applied in online mode, this ultimately leads to a large latency. One approach to reducing the latency is to break the source utterance into chunks and perform the translation on each chunk.

In this paper, we follow the incremental decoding framework described by Liu et al. (2020a). We break the input utterance into small fixed-size chunks and decode each time after we receive a new chunk. After each decoding step, we identify a stable part of the hypothesis using *stable hypothesis detection*. The stable part is sent to the user ("committed" in the following) and is no longer changed afterward (i.e., no retranslation).[2] Our current implementation assumes that the whole speech input fits into memory, in other words, we are only adding new chunks as they are arriving. This simplification is possible because the evaluation of the shared task is performed on segmented input, on individual utterances. With each newly arrived input chunk, the decoding starts with forced decoding of the already committed tokens and continues with beam search decoding.

[2]This is a requirement for the evaluation in the Simultaneous Speech Translation task at IWSLT 2022.

3.2 Chunk Size

Speech recognition and translation use chunking for simultaneous inference with various chunk sizes ranging from 300 ms to 2 seconds (Liu, 2020; Nguyen et al., 2021) although the literature suggests that the turn-taking in conversational speech is shorter, around 200 ms (Levinson and Torreira, 2015). We investigate different chunk sizes in combination with various stable hypothesis detection strategies. As we document later, the chunk size is the principal factor that controls the quality-latency trade-off.

3.3 Stable Hypothesis Detection

Committing hypotheses from incomplete input presents a possible risk of introducing errors. To reduce the instability and trade time for quality, we employ a *stable hypothesis detection*. Formally, we define a function $prefix(W)$ that, given a set of hypotheses (i.e., W^c_{all} if we want to consider the whole beam or W^c_{best} for the single best hypothesis obtained during the beam search decoding of the c-th chunk), outputs a stable prefix. We investigate several functions:

Hold-n (Liu et al., 2020a) Hold-n strategy selects the best hypothesis in the beam and deletes the last n tokens from it:

$$\text{prefix}(W^c_{best}) = W_{0:\max(0,|W|-n)}, \quad (1)$$

where W^c_{best} is the best hypothesis obtained in the beam search of c-th chunk. If the hypothesis has only n or fewer tokens, we return an empty string.

LA-n Local agreement (Liu et al., 2020a) displays the agreeing prefixes of the two consecutive chunks. Unlike the hold-n strategy, the local agreement does not offer any explicit quality-latency trade-off. We generalize the strategy to take the agreeing prefixes of n consecutive chunks.

During the first $n-1$ chunks, we do not output any tokens. From the n-th chunk on, we identify the longest common prefix of the best hypothesis of the n consecutive chunks:

$$\text{prefix}(W^c_{best}) = \begin{cases} \varnothing, & \text{if } c < n, \\ \text{LCP}(W^{c-n+1}_{best}, ..., W^c_{best}), & \text{otherwise,} \end{cases} \quad (2)$$

where $LCP(\cdot)$ is longest common prefix of the arguments.

SP-n Shared prefix (Nguyen et al., 2021) strategy displays the longest common prefix of all the items in the beam of a chunk. Similarly to the LA-n strategy, we propose a generalization to the longest common prefix of all items in the beams of the n consecutive chunks:

$$\text{prefix}(W^c_{all}) = \begin{cases} \varnothing, & \text{if } c < n, \\ \text{LCP}(W^{c-n+1}_{\text{beam } 1...B}, ..., W^c_{\text{beam } 1...B}), & \text{otherwise,} \end{cases} \quad (3)$$

i.e., all beam hypotheses $1, ..., B$ (where B is the beam size) of all chunks $c - n + 1, ..., c$.

3.4 Initial Wait

The limited context of the early chunks might result in an unstable hypothesis and an emission of erroneous tokens. The autoregressive nature of the model might cause further performance degradation in later chunks. One possible solution is to use longer chunks, but it inevitably leads to a higher latency throughout the whole utterance. To mitigate this issue, we explore a lengthening of the first chunk. We call this strategy an initial wait.

4 Experiments Setup

In this section, we describe the onlinization experiments.

4.1 Evaluation Setup

We use the SimulEval toolkit (Ma et al., 2020). The toolkit provides a simple interface for evaluation of simultaneous (speech) translation. It reports the quality metric BLEU (Papineni et al., 2002; Post, 2018) and latency metrics Average Proportion (AP, Cho and Esipova 2016), Average Lagging (AL, Ma et al. 2019), and Differentiable Average Lagging (DAL, Cherry and Foster 2019) modified for speech source.

Specifically, we implement an `Agent` class. We have to implement two important functions: `policy(state)` and `predict(state)`, where `state` is the state of the agent (e.g., read processed input, emitted tokens, ...). The policy function returns the action of the agent: (1) `READ` to request more input, (2) `WRITE` to emit new hypothesis tokens.

We implement the `policy` as specified in Algorithm 1. The default action is `READ`. If there is a new chunk, we perform the inference and use the $prefix(W^c)$ function to find the stable prefix. If there are new tokens to display (i.e., $|prefix(W^c)| > |prefix(W^{c-1})|$), we return the `WRITE` action. As soon as our agent emits an end-of-sequence (EOS) token, the inference of the utterance is finished by the SimulEval. We noticed that our model was emitting the EOS token quite often, especially in the early chunks. Hence, we ignore the EOS if returned by our model and continue the inference until the end of the source.[3]

Algorithm 1 Policy function

```
Require: state
  if state.new_input > chunk_size then
      hypothesis ← predict(state)
      if |hypothesis| > 0 then
          return WRITE
      end if
  end if
  return READ
```

4.2 Speech Translation Models

In our experiments, we use two different models. First, we do experiments with a monolingual *Model A*, then for the submission, we use a multilingual and more robust *Model B*.[4]

Model A is the KIT IWSLT 2020 model for the Offline Speech Translation task. Specifically, it is an end-to-end English to German Transformer model with relative attention. For more described description, refer to Pham et al. (2020b).

4.2.1 Multilingual Model

For the submission, we use a multilingual Model B. We construct the SLT architecture with the encoder based on the wav2vec 2.0 (Baevski et al., 2020) and the decoder based on the autoregressive language model pretrained with mBART50 (Tang et al., 2020).

wav2vec 2.0 is a Transformer encoder model which receives raw waveforms as input and generates high-level representations. The architecture consists of two main components: first, a

[3] This might cause an unnecessary increase in latency, but it might be partially prevented by voice activity detection.

[4] We also did experiments with a dedicated English-German model similar to Model B (i.e., based on wav2vec and mBART), but it performed worse both in offline and online setting compared to the multilingual version.

convolution-based *feature extractor* downsamples long audio waveforms into features that have similar lengths with spectrograms.After that, a deep Transformer encoder uses self-attention and feed-forward neural network blocks to transform the features without further downsampling.

During the self-supervised training process, the network is trained with a contrastive learning strategy (Baevski et al., 2020), in which the already downsampled features are randomly masked and the model learns to predict the quantized latent representation of the masked time step.

During the supervised learning step, we freeze the feature extraction weights to save memory since the first layers are among the largest ones. We fine-tune all of the weights in the Transformer encoder. Moreover, to make the model more robust to the fluctuation in absolute positions and durations when it comes to audio signals,we added the relative position encodings (Dai et al., 2019; Pham et al., 2020a) to alleviate this problem.[5]

Here we used the same pretrained model with the speech recognizer, with the large architecture pretrained with $53k$ hours of unlabeled data.

mBART50 is an encoder-decoder Transformer-based language model. During training, instead of the typical language modeling setting of predicting the next word in the sequence, this model is trained to reconstruct a sequence from its noisy version (Lewis et al., 2019) and later extended to a multilingual version (Liu et al., 2020b; Tang et al., 2020) in which the corpora from multiple languages are combined during training. mBART50 is the version that is pretrained on 50 languages.

The mBART50 model follows the Transformer encoder and decoder (Vaswani et al., 2017). During fine-tuning, we combine the mBART50 decoder with the wav2vec 2.0 encoder, where both encoder and decoder know one modality. The cross-attention layers connecting the decoder with the encoder are the parts that require extensive fine-tuning in this case, due to the modality mismatch between pretraining and fine-tuning.

Finally, we use the model in a multilingual setting, i.e., for English to Chinese, German, and Japanese language pairs by training on the combination of the datasets. The mBART50 vocabulary contains language tokens for all three languages and can be used to control the language output (Ha et al., 2016).

For more details on the model refer to Pham et al. (2022).

[5]This has the added advantage of better generalization in situations where training and testing data are segmented differently.

4.3 Test Data

For the onlinization experiments, we use MuST-C (Cattoni et al., 2021) `tst-COMMON` from the v2.0 release. We conduct all the experiments on the English-German language pair.

5 Experiments and Results

In this section, we describe the experiments and discuss the results.

5.1 Chunks Size

We experiment with chunk sizes of 250 ms, 500 ms, 1s, and 2 s. We combine the sizes of the chunks with different partial hypothesis selection strategies. The results are shown in Figure 1.

The results document that the chunk size parameter has a stronger influence on the trade-off than different prefix strategies. Additionally, this enables constant trade-off strategies (e.g., LA-2) to become flexible.

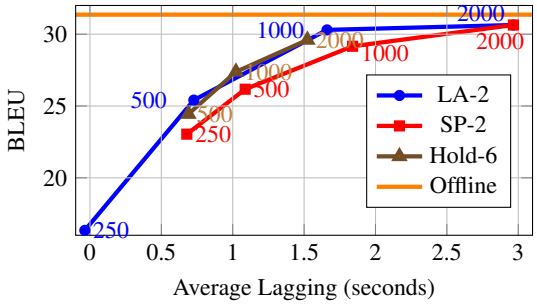

Figure 1: Quality-latency trade-off of different chunk sizes combined with different stable hypothesis detection strategies. The number next to the marks indicates chunk size in milliseconds.

5.2 Stable Hypothesis Detection Strategies

We experiment with three strategies: hold-n (withholds last n tokens), shared prefix (SP-n; finds the longest common prefix of all beams in n consecutive chunks) and local agreement (LA-n; finds the longest common prefix of the best hypothesis in n consecutive chunks). For hold-n, we select $n = 3, 6, 12$; for SP-n, we select $n = 1, 2$ ($n = 1$ corresponds to the strategy by Nguyen et al. (2021)); for LA-n we select $n = 2, 3, 4$ ($n = 2$

corresponds to the strategy by Liu et al. (2020a)). The results are in Figures 2 and 3.

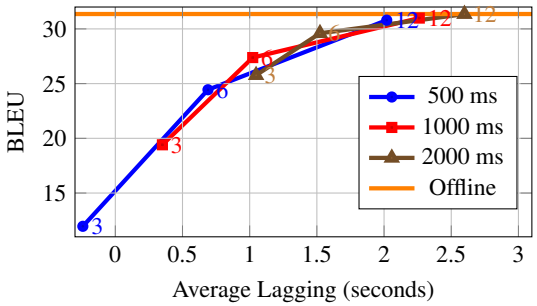

Figure 2: Quality-latency trade-off of hold-n strategy with different values of n. The number next to the marks indicates n. Colored lines connect results with equal chunk size.

Hold-n The results suggest (see Figure 2) that the hold-n strategy can use either n or chunk size to control the quality-latency trade-off with equal effect. The only exception seems to be too low $n <= 3$, which slightly underperforms the options with higher n and shorter chunk size.

Local agreement (LA-n) The local agreement seems to outperform all other strategies (see Figure 3). LA-n for all n follows the same quality-latency trade-off line. The advantage of LA-2 is in reduced computational complexity compared to the other LA-n strategies with $n > 2$.

Shared prefix (SP-n) SP-1 strongly underperforms other strategies in quality (see Figure 3). While the SP-1 strategy performs well in the ASR task (Nguyen et al., 2021), it is probably too lax for the speech translation task. The generalized and more conservative SP-2 performs much better. Although, the more relaxed LA-2, which considers only the best item in the beam, has a better quality-latency trade-off curve than the more conservative SP-2.

5.3 Initial Wait

As we could see in Section 5.1, the shorter chunk sizes tend to perform worse. One of the reasons might be the limited context of the early chunks.[6] To increase the early context, we prolong the first chunk to 2 seconds.

The results are in Table 1. We see a slight (0.3 BLEU) increase in quality for a chunk size of 250 ms, though the initial wait does not improve the BLEU and a considerable increase in the latency.

[6]If we translated a non-pre-segmented input, this problem would be limited only onetime to the beginning of the input.

Initial wait	Chunk size	BLEU	AL	AP	DAL
0	250	16.34	-35.97	0.66	1435.06
	500	25.40	727.55	0.73	1791.21
	1000	30.29	1660.59	0.83	2662.18
2000	250	16.60	358.35	0.74	2121.54
	500	25.42	952.15	0.77	2142.53
	1000	30.29	1654.77	0.83	2657.48

Table 1: Quality-latency trade-off of the LA-2 strategy with and without the initial wait.

The performance seems to be influenced mainly by the chunk size. The reason for smaller chunks' under-performance might be caused by (1) acoustic uncertainty towards the end of a chunk (e.g., words often get cut in the middle), or (2) insufficient information difference between two consecutive chunks.

This is supported by the observation in Figure 3. Increasing the number of consecutive chunks (i.e., increasing the context for the decision) considered in the local agreement strategy (LA-$2, 3, 4$), improves the quality, while it adds latency.

5.4 Negative Average Lagging

Interestingly, we noticed that some of the strategies achieved negative average lagging (e.g., LA-2 in Section 5.1) with a chunk size of 250 ms has AL of -36 ms). After a closer examination of the outputs, we found that the negative AL is in utterances where the hypothesis is significantly longer than the reference. Recall the average latency for speech input defined by Ma et al. (2020):

$$\text{AL}_{\text{speech}} = \frac{1}{\tau'(|\mathbf{X}|)} \sum_{i=1}^{\tau'(|\mathbf{X}|)} d_i - d_i^*, \qquad (4)$$

where $d_i = \sum_{k=1}^{j} T_k$, j is the index of raw audio segment that has been read when generating y_i, T_k is duration of raw audio segment, $\tau'(|\mathbf{X}|) = \min\{i | d_i = \sum_{j=1}^{|\mathbf{X}|} T_j\}$ and d_i^* are the delays of an ideal policy:

$$d_i^* = (i-1) \times \sum_{j=1}^{|\mathbf{X}|} T_j \; / |\mathbf{Y}^*|, \qquad (5)$$

where $\mathbf{Y}^*$ is reference translation.

If the hypothesis is longer than the reference, then $d_i^* > d_i$, making the sum argument in Equation (4) negative. On the other hand, if we use the length of the hypothesis instead, then a shorter

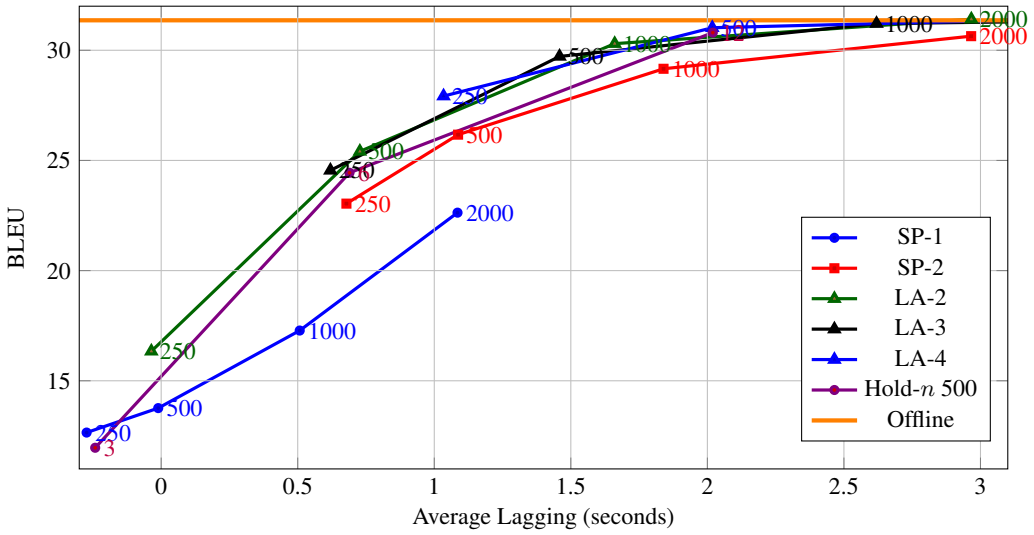

Figure 3: Quality-latency trade-off of shared prefix (SP-n) and local agreement (LA-n) with different n and chunk size.

hypothesis would benefit.[7] We, therefore, suggest using the maximum of both to prevent the advantage of either a shorter or a longer hypothesis:

$$d_i^* = (i-1) \times \sum_{j=1}^{|\mathbf{X}|} T_j \; / max(|\mathbf{Y}|, |\mathbf{Y}^*|). \quad (6)$$

6 Submitted System

In this section, we describe the submitted system. We follow the allowed training data and pretrained models and therefore our submission is *constrained* (see Section 4.2.1 for model description).

For stable hypothesis detection, we decided to use the local agreement strategy with $n = 2$. As shown in Section 5.2, the LA-2 has the best latency-quality trade-off along with other LA-n strategies. To achieve the different latency regimes, we use various chunk sizes, depending on the language pair. We decided not to use larger $n > 2$ to control the latency, as it increases the computation complexity while having the same effect as using a different chunk size. The results on MuST-C tst-COMMON are in Table 2. The quality-latency trade-off is in Figure 4.

From Table 2 and Figure 4, we can see that the proposed method works well on two different models and various language pairs. We see that an improvement in the offline model (offline BLEU of 31.36 and 33.14 for Model A and B, respectively) leads to improvement in the online regime.

[7] Ma et al. (2019) originally used the hypothesis length in the Equation (5) and then Ma et al. (2020) suggested to use the reference length instead.

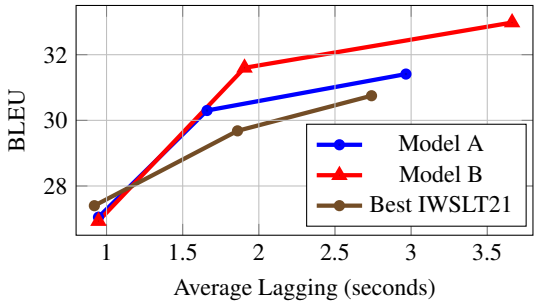

Figure 4: Quality-latency trade-off on English-German tst-COMMON of our two models: a dedicated English-German model trained from scratch (Model A) and a multilingual model based on wav2vec and mBART (Model B). We also include the best IWSLT 2021 system (USTC-NELSLIP (Liu et al., 2021)).

Finally, we see that our method beats the best IWSLT 2021 system (USTC-NELSLIP (Liu et al., 2021)) in medium and high latency regimes using both models (i.e., a model trained from scratch and a model based on pretrained wav2vec and mBART), and is almost on par in the low latency regime (Model A is losing 0.35 BLEU and Model B is losing 0.47 BLEU).

6.1 Computationally Aware Latency

In this paper, we do not report any computationally aware metrics, as our implementation of Transformers is slow. Later, we implemented the same onlinization approach using wav2vec 2.0 and mBART from Huggingface Transformers (Wolf et al., 2020). The new implementation reaches faster than real-time inference speed.

Model	Language pair	Latency regime	Chunk size	BLEU	AL	AP	DAL
Best IWSLT21 system	En-De	Low	-	27.40	920	0.68	1420
		Medium	-	29.68	1860	0.82	2650
		High	-	30.75	2740	0.90	3630
Model A	En-De	Low	600	27.05	947	0.76	1993
		Medium	1000	30.30	1660	0.84	2662
		High	2000	31.41	2966	0.93	3853
		Offline	-	31.36	5794	1.00	5794
Model B	En-De	Low	500	26.93	945	0.77	2052
		Medium	1000	31.60	1906	0.86	2945
		High	2500	32.98	3663	0.96	4452
		Offline	-	33.14	5794	1.00	5794
	En-Ja	Low	1000	16.84	2452	0.90	3212
		Medium	2400	16.99	3791	0.97	4296
		High	3000	16.97	4140	0.98	4536
		Offline	-	16.88	5119	1.00	5119
	En-Zh	Low	800	23.69	1761	0.85	2561
		Medium	1500	24.29	2788	0.93	3500
		High	2500	24.56	3669	0.97	4212
		Offline	-	24.54	5119	1.00	5119

Table 2: Results of the older model used for the experiments (Model A) and the submitted system (Model B) on the MuST-C v2 tst-COMMON. We also include the best IWSLT 2021 system (USTC-NELSLIP (Liu et al., 2021)).

7 Conclusion

In this paper, we reviewed onlinization strategies for end-to-end speech translation models. We identified the optimal stable hypothesis detection strategy and proposed two separate ways of the quality-latency trade-off parametrization. We showed that the onlinization of the offline models is easy and performs almost on par with the offline run. We demonstrated that an improvement in the offline model leads to improved online performance. We also showed that our method outperforms a dedicated simultaneous system. Finally, we proposed an improvement in the average latency metric.

Acknowledgments

This work has received support from the project "Grant Schemes at CU" (reg. no. CZ.02.2.69/0.0/0.0/19_073/0016935), the grant 19-26934X (NEUREM3) of the Czech Science Foundation, the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No 825460 (ELITR), and partly supported by a Facebook Sponsored Research Agreement "Language Similarity in Machine Translation".

References

Antonios Anastasopoulos, Luisa Bentivogli, Marcely Z. Boito, Ondřej Bojar, Roldano Cattoni, Anna Currey, Georgiana Dinu, Kevin Duh, Maha Elbayad, Marcello Federico, Christian Federmann, Hongyu Gong, Roman Grundkiewicz, Barry Haddow, Benjamin Hsu, Dávid Javorský, Věra Kloudová, Surafel M. Lakew, Xutai Ma, Prashant Mathur, Paul McNamee, Kenton Murray, Maria Nădejde, Satoshi Nakamura, Matteo Negri, Jan Niehues, Xing Niu, Juan Pino, Elizabeth Salesky, Jiatong Shi, Sebastian Stüker, Katsuhito Sudoh, Marco Turchi, Yogesh Virkar, Alex Waibel, Changhan Wang, and Shinji Watanabe. 2022. FINDINGS OF THE IWSLT 2022 EVALUATION CAMPAIGN. In *Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022)*, Dublin, Ireland. Association for Computational Linguistics.

Antonios Anastasopoulos, Ondřej Bojar, Jacob Bremerman, Roldano Cattoni, Maha Elbayad, Marcello Federico, Xutai Ma, Satoshi Nakamura, Matteo Negri, Jan Niehues, Juan Pino, Elizabeth Salesky, Sebastian Stüker, Katsuhito Sudoh, Marco Turchi, Alexander Waibel, Changhan Wang, and Matthew Wiesner. 2021. FINDINGS OF THE IWSLT 2021 EVALUATION CAMPAIGN. In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 1–29, Bangkok, Thailand (online). Association for Computational Linguistics.

Ebrahim Ansari, Amittai Axelrod, Nguyen Bach, Ondřej Bojar, Roldano Cattoni, Fahim Dalvi, Nadir Durrani, Marcello Federico, Christian Federmann, Jiatao Gu, Fei Huang, Kevin Knight, Xutai Ma, Ajay Nagesh, Matteo Negri, Jan Niehues, Juan Pino, Elizabeth Salesky, Xing Shi, Sebastian Stüker, Marco Turchi, Alexander Waibel, and Changhan Wang. 2020. FINDINGS OF THE IWSLT 2020 EVALUATION CAMPAIGN. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 1–34, Online. Association for Computational Linguistics.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed,

and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in Neural Information Processing Systems*, 33:12449–12460.

Parnia Bahar, Patrick Wilken, Mattia A. Di Gangi, and Evgeny Matusov. 2021. Without further ado: Direct and simultaneous speech translation by AppTek in 2021. In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 52–63, Bangkok, Thailand (online). Association for Computational Linguistics.

Roldano Cattoni, Mattia Antonino Di Gangi, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2021. Must-c: A multilingual corpus for end-to-end speech translation. *Computer Speech & Language*, 66:101155.

Colin Cherry and George Foster. 2019. Thinking slow about latency evaluation for simultaneous machine translation. *arXiv preprint arXiv:1906.00048*.

Kyunghyun Cho and Masha Esipova. 2016. Can neural machine translation do simultaneous translation? *arXiv preprint arXiv:1606.02012*.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Maha Elbayad, Ha Nguyen, Fethi Bougares, Natalia Tomashenko, Antoine Caubrière, Benjamin Lecouteux, Yannick Estève, and Laurent Besacier. 2020. ON-TRAC consortium for end-to-end and simultaneous speech translation challenge tasks at IWSLT 2020. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 35–43, Online. Association for Computational Linguistics.

Alex Graves. 2012. Sequence transduction with recurrent neural networks. *arXiv preprint arXiv:1211.3711*.

Thanh-Le Ha, Jan Niehues, and Alexander Waibel. 2016. Toward multilingual neural machine translation with universal encoder and decoder. In *Proceedings of the 13th International Workshop on Spoken Language Translation (IWSLT 2016)*, Seattle, USA.

Hou Jeung Han, Mohd Abbas Zaidi, Sathish Reddy Indurthi, Nikhil Kumar Lakumarapu, Beomseok Lee, and Sangha Kim. 2020. End-to-end simultaneous translation system for IWSLT2020 using modality agnostic meta-learning. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 62–68, Online. Association for Computational Linguistics.

Sathish Indurthi, Houjeung Han, Nikhil Kumar Lakumarapu, Beomseok Lee, Insoo Chung, Sangha Kim, and Chanwoo Kim. 2020. End-end speech-to-text translation with modality agnostic meta-learning. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7904–7908.

Muntsin Kolss, Stephan Vogel, and Alex Waibel. 2008. Stream decoding for simultaneous spoken language translation. In *Ninth Annual Conference of the International Speech Communication Association*.

Stephen C Levinson and Francisco Torreira. 2015. Timing in turn-taking and its implications for processing models of language. *Frontiers in psychology*, 6:731.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Dan Liu, Mengge Du, Xiaoxi Li, Yuchen Hu, and Lirong Dai. 2021. The USTC-NELSLIP systems for simultaneous speech translation task at IWSLT 2021. In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 30–38, Bangkok, Thailand (online). Association for Computational Linguistics.

Danni Liu. 2020. Low-latency end-to-end speech recognition with enhanced readability. Master's thesis, Maastricht University.

Danni Liu, Gerasimos Spanakis, and Jan Niehues. 2020a. Low-Latency Sequence-to-Sequence Speech Recognition and Translation by Partial Hypothesis Selection. In *Proc. Interspeech 2020*, pages 3620–3624.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020b. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Mingbo Ma, Liang Huang, Hao Xiong, Renjie Zheng, Kaibo Liu, Baigong Zheng, Chuanqiang Zhang, Zhongjun He, Hairong Liu, Xing Li, et al. 2019. Stacl: Simultaneous translation with implicit anticipation and controllable latency using prefix-to-prefix framework. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3025–3036.

Xutai Ma, Mohammad Javad Dousti, Changhan Wang, Jiatao Gu, and Juan Pino. 2020. Simuleval: An evaluation toolkit for simultaneous translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 144–150.

Thai-Son Nguyen, Sebastian Stüker, and Alex Waibel. 2021. Super-Human Performance in Online Low-Latency Recognition of Conversational Speech. In *Proc. Interspeech 2021*, pages 1762–1766.

Jan Niehues, Thai Son Nguyen, Eunah Cho, Thanh-Le Ha, Kevin Kilgour, Markus Müller, Matthias Sperber, Sebastian Stüker, and Alex Waibel. 2016. Dynamic Transcription for Low-Latency Speech Translation. In *Proc. Interspeech 2016*, pages 2513–2517.

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.

Ngoc-Quan Pham, Thanh-Le Ha, Tuan-Nam Nguyen, Thai-Son Nguyen, Elizabeth Salesky, Sebastian Stüker, Jan Niehues, and Alex Waibel. 2020a. Relative Positional Encoding for Speech Recognition and Direct Translation. In *Proc. Interspeech 2020*, pages 31–35.

Ngoc-Quan Pham, Tuan-Nam Nguyen, Thai-Binh Nguyen, Danni Liu, Carlos Mullov, Jan Niehues, and Alexander Waibel. 2022. Effective combination of pretrained models - KIT@IWSLT2022. In *Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022)*, Dublin, Ireland. Association for Computational Linguistics.

Ngoc-Quan Pham, Felix Schneider, Tuan Nam Nguyen, Thanh-Le Ha, Thai-Son Nguyen, Maximilian Awiszus, Sebastian Stüker, and Alex Waibel. 2020b. Kit's iwslt 2020 slt translation system. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 55–61.

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.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning. *arXiv preprint arXiv:2008.00401*.

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.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.