IMPROVING STABILITY IN SIMULTANEOUS SPEECH TRANSLATION: A REVISION-CONTROLLABLE DECODING APPROACH

Junkun Chen, Jian Xue, Peidong Wang, Jing Pan, Jinyu Li

Microsoft

ABSTRACT

Simultaneous Speech-to-Text translation serves a critical role in real-time crosslingual communication. Despite the advancements in recent years, challenges remain in achieving stability in the translation process, a concern primarily manifested in the flickering of partial results. In this paper, we propose a novel revision-controllable method designed to address this issue. Our method introduces an allowed revision window within the beam search pruning process to screen out candidate translations likely to cause extensive revisions, leading to a substantial reduction in flickering and, crucially, providing the capability to completely eliminate flickering. The experiments demonstrate the proposed method can significantly improve the decoding stability without compromising substantially on the translation quality.

***Index Terms*—** Flickering reduction, simultaneous speech translation, decoding stability, beam search

1. INTRODUCTION

Simultaneous Speech-to-Text translation (ST) incrementally translates speech in a source language speech into text in a target language, and has found wide-ranging applications in numerous crosslingual communication scenarios such as international travel and multinational conferences.

Unlike the full-sentence translation, which translates upon the cessation of speech segments, and provides a complete translation for an entire segment. Simultaneous ST requires the generation of intermediate translations as the speech continues. These partial results are of critical importance, enabling the audience to keep pace with the content of the speaker's discourse in real time, fostering immediate comprehension and engagement.

In recent years, the end-to-end (E2E) approach has surpassed conventional cascaded methods in terms of performance [1, 2]. Notably, the implementation of Transducer models for the adaptive simultaneous translation for streaming speech has significantly enhanced translation quality [3, 4]. Despite these advancements, the stability issue remains unaddressed in this task.

As the speech continues, a simultaneous ST system does not inherently guarantee to append new words to the previous partial result. As shown in Figure 1(a), words previously displayed can be removed or altered. This instability in the partial results can lead to frequent alterations on screen, causing the results to flicker. While permitting revisions has the potential to improve translation quality, this flickering creates an unfavorable user experience and can be distracting [5, 6]. It causes discomfort among audience members, who might consequently lose track of the content. Given the reordering nature between different languages [7, 8], the experience with flickering is substantially worse than that of Automatic Speech Recognition (ASR) tasks, which maintain a monotonic alignment.

Furthermore, simultaneous ST is usually succeeded by an incremental Text-To-Speech (TTS) system that synthesizes the text into speech in the target language [9]. Since the synthesized speech display cannot be retroactively altered, flickering poses significant challenges.

In contrast to the ASR task, which invariably aims for

Source Transcription	*měiguó zhōng xī bù yǒu hěnduō guójiā gōngyuán* 美国 的 西 部 有 很多 国家 公园 *USA 's west area have many national parks*
Translation-Ref	**there are many national parks in western US**
(a) E2E Streaming Translation	*(audio and segment start)* $[t_1]$ **American** $[t_2]$ **Western US** $[t_3]$ **Western US has many** $[t_3]$ **there are many national parks in western US** *(audio and segment end)*
(b) Revision-Free Decoding	*(audio and segment start)* $[t_1]$ **American** $[t_2]$ **American west** $[t_3]$ **American west has many** $[t_3]$ **American west has many national parks** *(audio and segment end)*

Fig. 1: A decoding example of E2E simultaneous ST. The provided source transcription represents the content of the source speech in Chinese, with the corresponding gloss also displayed. Text in **blue** denote newly generated translation. The standard approach showcases the possibility for intermediate translation to flicker with continuous speech input. In contrast, our proposed revision-free decoding method strives to maintain the intermediate translation unrevised.

979-8-3503-0689-7/23/$31.00 ©2023 IEEE

a single optimal outcome as the desired recognition result. The ST can accommodate multiple valid translations for a single input. For instance, as shown in Fig. 1, both "*American midwest*" and "*West central US*" can serve as suitable translations for "美国的中西部". Furthermore, the flexibility of languages allows for multiple equivalent expressions [10]. Therefore, not all revisions are necessary. To maintain the stability of partial results, a continuous translation strategy can be implemented, which builds upon the previously generated prefix. As illustrated in Fig. 1(b), a revision-free decoding approach, which refrains from modifying generated partial results, can also yield translations of considerable quality.

While greedy decoding is widely utilized in simultaneous translation to guarantee stability [11], it often compromises the overall translation quality [12]. In this work, we propose two strategies aimed at mitigating the flickering issues observed in simultaneous ST. We identify the root cause of flickering in the ranking process of beam search decoding. It presents a potential avenue for reducing the frequency of intermediate translation commitments and thereby preventing unnecessary revisions. Furthermore, to fundamentally address the flickering issue with beam search, we introduce a novel revision-controllable approach that actively manages revisions during the translation decoding process. Our primary idea is to modify the beam search pruning process through the introduction of an allowed revision window. It can filter out candidates that may induce extensive revisions. Our method can completely prevent flickering during decoding with only a minor reduction in translation quality.

By adjusting the allowed revision window, our proposed method is capable of achieving performance comparable to existing methods in terms of translation quality, while significantly enhancing both latency and stability. This dual strategy presents a promising solution for enhancing the performance and user experience of simultaneous ST.

2. PRELIMINARY

We first briefly review end-to-end simultaneous speech translation to set up the notations.

2.1. End-to-End Speech Translation

Regardless of the specific model architecture employed, the novel paradigm of E2E ST delineates an objective: to transform a speech feature sequence $\mathbf{x} = (x_1, x_2, ..., x_T)$ into a series of text tokens $\mathbf{y}$ in a different language, where each x_i represents the frame-level feature with a certain duration.

The conventional cascaded system uses an ASR model to convert speech to text in the source language, which is subsequently fed into a machine translation (MT) model to get the translation in the target language. Unlike this method, the E2E ST model incorporates a singular, integrated model which does not need an intermediate recognized result,

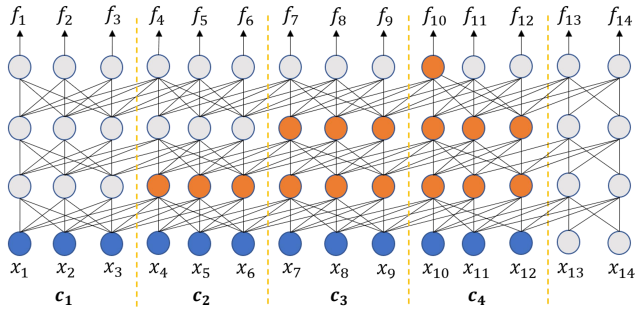


Fig. 2: TT encoding at position f_{10}, utilizing an attention mask. The process is characterized by a specified chunk size of 3 and the number of left chunk 1.

thereby alleviating the issue of error propagation [13]. The model can be formalized as:

$$p_{\text{full}}(\mathbf{y} \mid \mathbf{x}; \boldsymbol{\theta}) = \prod_{t=1}^{|\mathbf{y}|} p(y_t \mid \mathbf{x}, \mathbf{y}_{<t}; \boldsymbol{\theta})$$

2.2. Simultaneous End-to-End Speech Translation with Transducer Model

Simultaneous E2E ST system translates concurrently with continuous source speech. Formally, the prediction of $\mathbf{y}$ can be defined as,

$$p_{\text{simul}}(\mathbf{y} \mid \mathbf{x}; \boldsymbol{\theta}) = \prod_{t=1}^{|\mathbf{y}|} p(y_t \mid \mathbf{x}_{<\tau(t)}, \mathbf{y}_{<t}; \boldsymbol{\theta})$$

where $\tau(t)$ denotes the timestamp to decode target token y_t.

Recently, the neural transducer model [14] presents a compelling fit for simultaneous E2E ST [3, 4]. Its design considers all potential alignments between speech and text throughout the training process, and it shows the capacity to adaptively translate speech into text in a streaming manner [15]. In [3], the authors proposed to leverage the Transformer-Transducer (T-T) [16, 17] for simultaneous ST. To realize the low-latency high-accuracy streaming T-T, they use the speech encoder design in [18], shown in Fig. 2.

To uphold low latency and minimize computational costs, the input speech frames $\mathbf{x}$ are segmented and sequentially fed into the encoder in distinctive chunks $\mathbf{c}$. Each chunk $\mathbf{c}_i$ comprises several speech frames, the quantity of which aligns with the chunk size u. The incorporation of the attention masks [18] facilitates processing in a chunk-wise streaming fashion. Every frame has a predetermined number of visible left chunks, and the size of the left reception field grows proportionally with the number of layers. This allows the model to leverage extensive historical information for improved performance while significantly reducing computational requirements compared to models that consider the entire history at each layer. Within a chunk, all frames can observe one another, but they cannot access frames in subsequent chunks.

2.3. Decoding with Beam Search

In order to achieve a fluent translation, the application of beam search is crucial during the decoding process. We denote B_i to be the beam at time step i, which is an ordered list with a beam size of b, and it expands to the next beam B_{i+1} with the same size:

$$\begin{aligned} B_0 =& [\langle \text{<s>}, p_{\text{simul}}(\text{<s>} \mid x_1; \boldsymbol{\theta})\rangle] \\ B_i =& \mathbf{top}^b(next(B_{i-1}, i)) \\ next(B, i) =& \{\langle \mathbf{y} \circ y_i, p_{\text{simul}}(y_i \mid \mathbf{x}_{\leq \tau(i)}, \mathbf{y}; \boldsymbol{\theta})\rangle \mid \\ & \langle \mathbf{y}, p\rangle \in B, y_i \in V\} \\ \hat{\mathbf{y}}_i =& \mathbf{top}^1(next(B_{i-1}, i))[0] \end{aligned}$$

The best hypothesis $\hat{\mathbf{y}}_i$ is usually used as the intermediate generated result [19]. Intermediate result revision happens when the top candidate $\hat{\mathbf{y}}_i$ in B_i is neither identical to nor a prefix of the top candidate $\hat{\mathbf{y}}_{i+1}$ in B_{i+1}.

3. METHODS

As the simultaneous ST model processes continuous speech, it generates intermediate translation results. In the context of online decoding utilizing beam search, the intermediate translation is typically represented by the best candidate $\hat{\mathbf{y}}$ in the beam. However, as discussed in Sec. 2.3, these best candidates may not always serve as the prefix for the subsequent translation. This discrepancy arises due to the inherent reranking property of the beam search algorithm, which can modify the candidate order based on their evolving scores as the decoding process advances.

3.1. Chunk Preservation

In conventional methods for text-based simultaneous MT, such as the widely-used wait-k method [11], the model commits a single target token each time it receives a new source token. However, this approach may not be the most efficient for Speech-to-Text translation, where the input granularity is at the frame level. Individually, frames often lack sufficient semantic information, and their length generally exceeds that of the target text sequence.

Adopting the chunk-based model, as detailed in the Sec. 2.2, allows the input to be processed chunk-by-chunk. In this approach, it is not necessary to commit results at the frame level. Committing at the chunk level is more logical given the characteristics of ST. This modification offers two primary advantages:

- It reduces the frequency of commitments, thereby preventing unnecessary revisions within each chunk.
- It streamlines the process for committing outputs, saving computational efforts and reducing system communication overhead.

Therefore, we proposed a method called Chunk Preservation (*CP*). For instance, as depicted in Fig. 2, the proposed method refrains from committing intermediate translations for individual frames such as f_{10}. Instead, it only commits the best candidates as the intermediate translation at the end frame of each chunk, such as f_{12} in the example. It can be formalized as,

$$\mathbf{commit}(\hat{\mathbf{y}}_i, u) = \begin{cases} \text{True} & \text{if } i \bmod u = 0 \\ \text{False} & \text{otherwise.} \end{cases}$$

By committing results at the chunk level rather than the frame level, the translation process aligns more closely with the natural input processing pattern of ST, greatly enhancing the decoding stability.

It is important to note that while this method modifies the commitment approach, it leaves the translation quality unaffected. The translation accuracy remains consistent with that achieved through standard frame-level commitment.

3.2. Revision Window Control

However, chunk preservation cannot fundamentally address the issue of revision. Given that revisions originate from the ranking process in beam search, we can maintain a beam where the revisions applied to subsequent translations from all candidates are kept within a specified limit. Thus, we design a revision-controllable decoding method, rooted in the beam search process, in which we incorporate a Revision Window (*RW*). This value regulates the maximum number of tokens that can be revised in subsequent decoding. Specifically, every time we commit the intermediate translation (at the end of each chunk), we prune the beam with this revision window control strategy in effect.

The proposed beam pruning method can be described as,

$$\mathbf{accept}(\mathbf{x}, \mathbf{y}, RW) = \begin{cases} \text{True}, & \text{if } \mathbf{x}_{:|\mathbf{y}|-RW} = \mathbf{y}_{:|\mathbf{y}|-RW} \\ \text{False}, & \text{otherwise} \end{cases}$$

$$B_i = \mathbf{top}^{b*}(next(B_{i-1}, i))$$
$$\forall \langle \mathbf{y}, p\rangle \in next(B_{i-1}, i),\ \mathbf{accept}(\mathbf{y}, \hat{\mathbf{y}}_i, RW).$$

In essence, all the surviving candidates within the beam must maintain an identical prefix, with the length being subject to the provided revision window. Otherwise, they will be pruned, regardless of their scores. *RW* denotes how many tokens at the end of the intermediate translation are permitted to be revised in the beam search process due to the progress and reranking stemming from the ongoing translation. An extreme case occurs when $RW = 0$. In this case, all candidates employ the best candidates as the prefix [1], ensuring that subsequent translations will not revise the previous intermediate translation. In scenarios that do not require strict controls on revision, *RW* can be adjusted to strike a balance between translation quality and stability.

[1] it is possible that fewer than b candidates survive.

4. EXPERIMENTS

4.1. Data and Model

Language	DE	ES	IT	FR	NL	ZH
Hours	15k	17k	16k	14k	5k	65k

Table 1: Statistics of training speech corpora for each source language.

To demonstrate the effectiveness of our proposed method, we conducted experiments on multiple translation directions.

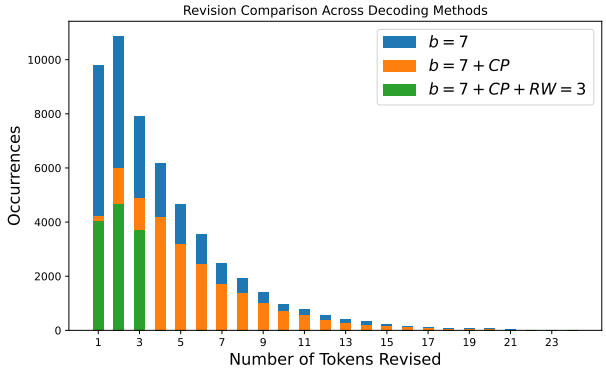
Following [20], we trained a multilingual ST model that effectively translates speech from various languages into English. In the experiment, we used a collection of anonymized internal speech corpora intended for ASR. The specific languages covered, along with their corresponding durations of training data, are shown in Table 1. The training translation references were annotated with a MT service. The model is constructed with a direct translation framework, allowing it to seamlessly process speech in a set of languages, specifically German (DE), Spanish (ES), Italian (IT), French (FR), Dutch (NL), and Chinese (ZH), into English (EN) without necessitating any language-specific configuration.

We adopt the Transformer-Transducer as the foundational architecture for our model, with a chunk size of 4 and masked attention, as detailed in Sec. 2, to enable the ability to process streaming input. We set the chunk size $u = 4$ (i.e., 4 frames per chunk and the frame span is 40ms) and set the left chunk value to 18. More specifically, the encoder consists of a Transformer architecture with 18 layers with 2048 hidden size, each having 8 attention heads with an attention dimension of 256. For the prediction network, we use a 2-layer stacked LSTM [21], with a hidden size of 1024, thus allowing for efficient sequence prediction. We set the embedding size to 320. The model is trained with AdamW optimizer [22] for 1.6 million steps.

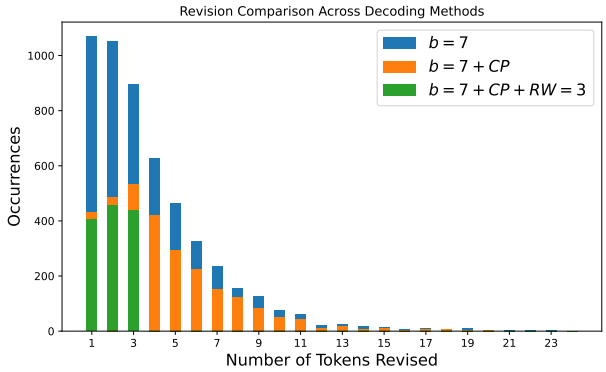
The efficacy of our proposed method is evaluated on the CoVoST2 [23] X→En translation set, with individual assessments conducted for each respective language pair. Our evaluation metric encompasses three core dimensions: translation quality, latency, and stability.

For translation quality, we report the case-sensitive detokenized BLEU using `sacreBLEU`[2] [24]. In terms of latency, we employ Average Lagging (AL) [11] in milliseconds. This crucial measure enables us to understand the real-time applicability of our method in practical scenarios. Given that the AL metric is traditionally intended for decoding processes wherein intermediate results remain unrevised, and typically employed for greedy decoding methodologies, we adjust its application for our research context. In this study, we conduct an offline evaluation of AL by analyzing the timestamps

[2]https://github.com/mjpost/sacreBLEU



(a) DE→EN



(b) NL→EN

Fig. 3: Revision count comparison across three decoding methods. This bar chart presents the comparison of decoding steps (y-axis) of the number of tokens been revised (x-axis) with three different decoding methods. It allows for a direct comparison of how often specific numbers of revisions occur in each method.

corresponding to each decoded token. Specifically, we identify the moment when a decoded token is finalized and subsequently remains unchanged for the rest of the decoding process. We leverage the index of the frame in which the intermediate result is committed to measure the latency (AL). This differs from using the end frame of each chunk, which represents real non-computation aware latency. Our chosen method is adopted to effectively account for the latency introduced by computational processes and display time. Finally, for assessing the stability of our method, we employ the metric of Normalize Erasure (NE) [25]. This metric quantifies the number of partial target tokens that are erased relative to each final target token.

4.2. Evaluation

Throughout the experimentation with our proposed method, we consistently employ a *beam* size $b = 7$. It is noteworthy that conventional beam search algorithms tend to favor

	DE → EN			ES → EN			IT → EN			FR → EN			NL → EN			ZH → EN		
	BLEU ↑	AL ↓	NE ↓	BLEU ↑	AL ↓	NE ↓	BLEU ↑	AL ↓	NE ↓	BLEU ↑	AL ↓	NE ↓	BLEU ↑	AL ↓	NE ↓	BLEU ↑	AL ↓	NE ↓
$b = 1$	19.55	1317	0.00	18.96	1239	0.00	17.94	1270	0.00	19.04	1112	0.00	21.17	1183	0.00	2.02	3140	0.00
$b = 7$	26.28	1057	1.49	26.68	1054	1.74	26.50	1052	1.59	26.30	1061	1.60	28.28	967	1.37	10.97	1418	2.56
+ *CP*	~	1110	1.00	~	1105	1.15	~	1106	1.08	~	1035	1.07	~	1045	0.92	~	1122	2.08
+ *RW* = 0	25.13	689	0.00	24.28	549	0.00	25.18	648	0.00	24.99	591	0.00	27.11	756	0.00	10.32	748	0.00
+ *RW* = 3	26.33	800	0.11	26.61	730	0.11	26.55	768	0.11	26.41	707	0.11	28.27	842	0.12	11.07	922	0.12

Table 2: Performance metrics on the CoVoST2 test set across various translation directions. ~ denotes that the value is the same as the one in the row above. Chunk preservation does not change the decoding results, it yields the same BLEU score as the standard frame-level beam search decoding.

shorter hypotheses, a characteristic often encountered in machine translation scenarios. To counterbalance this inherent bias towards brevity, we incorporate a *Word Reward* [26] parameter set to 1 in beam search where revision window control is not employed.

The evaluation results are shown in Table 2. It is evident that utilizing a beam size $b = 1$ ensures no revision in decoding. However, it markedly compromises the performance. This approach typically generates hypotheses shorter than expected, primarily because shorter hypotheses tend to have better scores. The availability of only a single candidate as the prefix throughout the entire decoding process hinders the growth of the sequence in subsequent decoding, given that transducer decoding follows a frame-level synchronized style, As a result, it incurred a severe brevity penalty, leading to both poor quality and increased latency. This underscores the critical role of beam search in decoding with transducer models, unlike in the case of sequence-to-sequence models. The use of chunk preservation in our model effectively mitigates flickering and improves stability, as indicated by the NE scores. Despite this minor latency increase, the gains in stability make this a beneficial trade-off.

The introduction of a controlled revision window plays a pivotal role in our method. In contrast to pruning the beam to a size of 1, our revision-controllable method is capable of maintaining multiple candidates in the beam as long as they do not introduce flickering beyond the given revision window in subsequent decoding. The method can fundamentally prevent unnecessary revisions and achieves enhanced latency (as reflected with $RW = 0$). Moreover, our model exhibits significant flexibility, facilitated by the adjustability of the revision window. With $RW = 3$, the model is able to achieve translation quality comparable to that of beam search without revision window control. And it still leads to significant enhancements in both latency and stability. Especially, the enhanced stability, indicated by the lower NE scores, ensures the model's output consistency, reinforcing the reliability of the translations produced. This clear improvement in both latency and stability, without compromising the translation quality, underlines the effectiveness of our proposed method.

4.3. Analysis on Revision

While our proposed methods yield a notable improvement, to assess the stability of decoding at a more granular level, we performed an analysis of the frequency count for each specified number of tokens revised during the decoding process.

As demonstrated in Figure 3, the chunk preservation method significantly mitigates the flickering issue for both DE→EN and NL→EN translation, leading to fewer revisions during decoding and thereby enhancing NE. Despite these improvements, chunk preservation is unable to prevent extreme cases of long-range revisions where a large number of previously generated tokens are revised, drastically undermining stability.

In contrast, our proposed revision-controllable method effectively counteracts this problem. By implementing an allowable revision window (*RW*), we establish an upper limit to the number of tokens that can be revised. This is accomplished via a novel pruning method within the beam search process (as detailed in Sec. 3.2), significantly bolstering the stability of the decoding process.

5. RELATED WORKS

The conventional approach to preventing flickering during the decoding process involves the use of greedy decoding, a technique frequently employed in text-based simultaneous MT [11, 27]. However, for transducer-based streaming decoding, this method proves unsatisfactory due to its inherent limitations on output quality. For re-translation-style simultaneous translation models, a biased beam search approach has been utilized to enforce the decoding of previously generated text as a prefix [25].

The most relevant study to our work is [28], the authors propose an altered approach to hypothesis selection within the beam during partial generation. Instead of always selecting the highest-ranked hypothesis from the beam, they introduce a method to select the partial result, sticking a balance between flickering, quality, and latency. This method is employed in steaming ASR not simultaneous ST, which contains long-distance reorderings. And it does not modify the beam search process.

6. CONCLUSION

In this work, we presented a simple and effective method to address the issue of decoding stability in E2E simultaneous ST, particularly, the flickering of partial translation results. We propose two methods for reducing flickering. First, we introduce a straightforward technique, called chunk preservation, which significantly reduces flickering while maintaining the translation quality. Second, we proposed a novel revision-controllable method that introduces an allowed revision window within the beam search pruning process. This approach effectively filters out candidate translations that could lead to extensive revisions, thereby significantly reducing flickering and enhancing the stability of the translation. Moreover, by limiting the maximum number of tokens that can be revised, our method successfully prevents extreme instances of instability, thereby significantly improving user experience. Importantly, these improvements were achieved without a significant compromise on translation quality.

7. REFERENCES

[1] Yi Ren, Jinglin Liu, Xu Tan, Chen Zhang, Tao Qin, Zhou Zhao, and Tie-Yan Liu, "Simulspeech: End-to-end simultaneous speech to text translation," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 3787–3796.

[2] Junkun Chen, Mingbo Ma, Renjie Zheng, and Liang Huang, "Direct simultaneous speech-to-text translation assisted by synchronized streaming asr," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 2021, pp. 4618–4624.

[3] Jian Xue, Peidong Wang, Jinyu Li, Matt Post, and Yashesh Gaur, "Large-scale streaming end-to-end speech translation with neural transducers," in *Proc. Interspeech*, 2022, pp. 3263–3267.

[4] Yun Tang, Anna Sun, Hirofumi Inaguma, Xinyue Chen, Ning Dong, Xutai Ma, Paden Tomasello, and Juan Pino, "Hybrid transducer and attention based encoder-decoder modeling for speech-to-text tasks," in *Proc. ACL*, 2023, pp. 12441–12455.

[5] Timo Baumann, Michaela Atterer, and David Schlangen, "Assessing and improving the performance of speech recognition for incremental systems," in *Proceedings of human language technologies: The 2009 annual conference of the north american chapter of the association for computational linguistics*, 2009, pp. 380–388.

[6] Moritz Stolte and Ulrich Ansorge, "Automatic capture of attention by flicker," *Attention, Perception, & Psychophysics*, vol. 83, pp. 1407–1415, 2021.

[7] Alexandra Birch, Phil Blunsom, and Miles Osborne, "A quantitative analysis of reordering phenomena," in *Proceedings of the Fourth Workshop on Statistical Machine Translation*. Association for Computational Linguistics, 2009, pp. 197–205.

[8] Fabienne Braune, Anita Gojun, and Alexander Fraser, "Long-distance reordering during search for hierarchical phrase-based smt," in *Proceedings of the Annual Conference of the European Association for Machine Translation (EAMT)*. Citeseer, 2012, pp. 28–30.

[9] Renjie Zheng, Mingbo Ma, Baigong Zheng, Kaibo Liu, Jiahong Yuan, Kenneth Church, and Liang Huang, "Fluent and low-latency simultaneous speech-to-speech translation with self-adaptive training," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, 2020, pp. 3928–3937.

[10] Philipp Koehn, *Statistical machine translation*, Cambridge University Press, 2009.

[11] Mingbo Ma, Liang Huang, Hao Xiong, Renjie Zheng, Kaibo Liu, Baigong Zheng, Chuanqiang Zhang, Zhongjun He, Hairong Liu, Xing Li, et al., "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*, 2019, pp. 3025–3036.

[12] Renjie Zheng, Mingbo Ma, Baigong Zheng, and Liang Huang, "Speculative beam search for simultaneous translation," 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, pp. 1395–1402.

[13] Matthias Sperber, Graham Neubig, Jan Niehues, and Alex Waibel, "Attention-passing models for robust and data-efficient end-to-end speech translation," *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 313–325, 2019.

[14] Alex Graves, "Sequence transduction with recurrent neural networks," *arXiv preprint arXiv:1211.3711*, 2012.

[15] Sara Papi, Peidong Wan, Junkun Chen, Jian Xue, Jinyu Li, and Yashesh Gaur, "Token-level serialized output training for joint streaming asr and st leveraging textual alignments," *arXiv preprint arXiv:2307.03354*, 2023.

[16] Ching-Feng Yeh, Jay Mahadeokar, Kaustubh Kalgaonkar, Yongqiang Wang, Duc Le, Mahaveer Jain,

Kjell Schubert, Christian Fuegen, and Michael L Seltzer, "Transformer-transducer: End-to-end speech recognition with self-attention," *arXiv preprint arXiv:1910.12977*, 2019.

[17] Qian Zhang, Han Lu, Hasim Sak, Anshuman Tripathi, Erik McDermott, Stephen Koo, and Shankar Kumar, "Transformer transducer: A streamable speech recognition model with transformer encoders and rnn-t loss," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7829–7833.

[18] Xie Chen, Yu Wu, Zhenghao Wang, Shujie Liu, and Jinyu Li, "Developing real-time streaming transformer transducer for speech recognition on large-scale dataset," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 5904–5908.

[19] Dong Yu and Lin Deng, *Automatic speech recognition*, vol. 1, Springer, 2016.

[20] Jian Xue, Peidong Wang, Jinyu Li, and Eric Sun, "A weakly-supervised streaming multilingual speech model with truly zero-shot capability," *arXiv preprint arXiv:2211.02499*, 2022.

[21] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[22] Ilya Loshchilov and Frank Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2018.

[23] Changhan Wang, Anne Wu, Jiatao Gu, and Juan Pino, "Covost 2 and massively multilingual speech translation," *Proc. Interspeech 2021*, pp. 2247–2251, 2021.

[24] Matt Post, "A call for clarity in reporting BLEU scores," in *Proceedings of the Third Conference on Machine Translation: Research Papers*, Belgium, Brussels, Oct. 2018, pp. 186–191, Association for Computational Linguistics.

[25] Naveen Arivazhagan, Colin Cherry, Isabelle Te, Wolfgang Macherey, Pallavi Baljekar, and George Foster, "Re-translation strategies for long form, simultaneous, spoken language translation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7919–7923.

[26] Wei He, Zhongjun He, Hua Wu, and Haifeng Wang, "Improved neural machine translation with smt features," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*. AAAI Press, 2016, pp. 151–157.

[27] Junkun Chen, Renjie Zheng, Atsuhito Kita, Mingbo Ma, and Liang Huang, "Improving simultaneous translation by incorporating pseudo-references with fewer reorderings," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 5857–5864.

[28] Antoine Bruguier, David Qiu, Trevor Strohman, and Yanzhang He, "Flickering reduction with partial hypothesis reranking for streaming asr," in *2022 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2023, pp. 38–45.