High-Fidelity Simultaneous Speech-To-Speech Translation

Tom Labiausse [1] **Laurent Mazaré** [1] **Edouard Grave** [1] **Alexandre Défossez** [1] **Neil Zeghidour** [1]

Abstract

We introduce Hibiki, a decoder-only model for simultaneous speech translation. Hibiki leverages a multistream language model to synchronously process source and target speech, and jointly produces text and audio tokens to perform speech-to-text and speech-to-speech translation. We furthermore address the fundamental challenge of *simultaneous* interpretation, which unlike its *consecutive* counterpart—where one waits for the end of the source utterance to start translating—adapts its flow to accumulate just enough context to produce a correct translation in real-time, chunk by chunk. To do so, we introduce a weakly-supervised method that leverages the perplexity of an off-the-shelf text translation system to identify optimal delays on a per-word basis and create aligned synthetic data. After supervised training, Hibiki performs adaptive, simultaneous speech translation with vanilla temperature sampling. On a French-English simultaneous speech translation task, Hibiki demonstrates state-of-the-art performance in translation quality, speaker fidelity and naturalness. Moreover, the simplicity of its inference process makes it compatible with batched translation and even real-time on-device deployment. We provide examples[1] as well as models and inference code.[2]

1. Introduction

We introduce Hibiki ("echo" in Japanese), a system for streaming and expressive speech-to-speech (S2ST) and speech-to-text (S2TT) translation. Most work in speech translation has focused on the offline (or *consecutive*) setting where the model has access to the full source utterance before it translates, as it provides useful context while fitting many use cases such as offline video dubbing. A more challenging setting is that of *simultaneous* translation, where translated speech is produced on-the-fly. This task requires a real-time decision-making process to evaluate whether enough context has been accumulated to reliably translate another chunk of speech. When cast as a machine learning problem, this endeavor presents additional challenges such as the lack of speech aligned at a chunk-level.

Hibiki is a decoder-only model which synchronously receives source speech and generates translated speech by leveraging a multistream architecture. In this context, nested global and local Transformers (Vaswani et al., 2017) jointly model two audio streams by predicting a hierarchy of text and audio tokens for each of them. At inference time, temperature sampling combined with a causal audio codec allows for streaming inputs and outputs. While this architecture was originally introduced by Défossez et al. (2024) for full-duplex spoken dialogue, we show how it provides a simple and convenient architecture for simultaneous speech translation. To train Hibiki, we generate synthetic parallel data by translating and resynthesizing the transcript of single-language audio. While this provides pairs of inputs and outputs aligned at the sequence level, this does not allow learning fine-grained alignments. We thus introduce "contextual alignment", a simple method based on the perplexity of an off-the-shelf machine translation system (Kudugunta et al., 2023) to derive word-level alignments. By then introducing proper silences into target speech, we can train Hibiki to adapt its flow in real-time, without the need for complex inference policies. Moreover, observing that training data varies widely in speaker similarity, we propose to label training examples with categories of speaker similarity, which avoids filtering the training data while allowing to favor high speaker similarity at inference with classifier-free guidance.

In a French-to-English translation task, Hibiki outperforms previous work in translation quality, speaker similarity and naturalness. We demonstrate how the simplicity of our inference process allows for translating hundreds of sequences in real-time on GPU, and how a distilled model can run in real-time on a smartphone. Human evaluations also show that Hibiki is the first model to provide an experience of interpretation close to human professionals. We will release our code, models, and a high quality 900 hours synthetic dataset.

[1]Kyutai, Paris, France. Correspondence to: Hibiki <hibiki@kyutai.org>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

[1]https://hf.co/spaces/kyutai/hibiki-samples

[2]https://github.com/kyutai-labs/hibiki

2. Related Work

2.1. End-to-end speech translation

Speech translation can be traced back to the early 1990s (Jain et al., 1991) with a first generation of systems that combined automatic speech recognition (ASR), machine translation (MT) and text-to-speech synthesis (TTS). While such cascaded approaches allowed for the growth of speech translation (Wahlster, 2000; Nakamura et al., 2006), they suffer from two main limitations. First, they are subject to compounding errors due to combining separately trained models. This motivated the merging of ASR and MT into a single speech-to-text translation (S2TT) model (Berard et al., 2016; Weiss et al., 2017; Wang et al., 2020; 2021b) that can provide inputs to a TTS model. However, a second limitation of cascaded systems remains: as the input speech goes through a text bottleneck, the non-linguistic information it carries—such as speaker identity or prosody—is lost and cannot be transferred to the output speech. End-to-end speech-to-speech translation (S2ST) (Jia et al., 2019; Lee et al., 2022a; Jia et al., 2022a; Rubenstein et al., 2023) addresses this issue by directly predicting target speech from source speech, allowing for retaining paralinguistic information, including voice identity. A notable aspect of most end-to-end S2ST models is that they leverage auxiliary text or phoneme translation tasks in training, that are then discarded (Jia et al., 2022a) or run in parallel (Zhang et al., 2024a) to the main speech translation task at inference. Hibiki performs end-to-end S2ST with voice transfer along with S2TT, but instead of running these tasks in parallel, Hibiki uses the predicted text as a scaffolding for speech generation at inference time. Moreover, since Hibiki predicts aligned speech and text tokens, it provides word-level timestamps in the target language.

2.2. Simultaneous speech translation

While the first attempts at simultaneous speech translation focused on speech-to-text (Ren et al., 2020; Ma et al., 2021; Zeng et al., 2021), Seamless (Barrault et al., 2023) and StreamSpeech (Zhang et al., 2024a) have introduced end-to-end simultaneous S2ST (Zeng et al., 2021; Barrault et al., 2023; Zhang et al., 2024a). Both systems predict discrete speech units with autoregressive models before decoding them to audio using a neural vocoder, and rely on a specific policy for inference. While StreamSpeech translates into a canonical voice, Seamless performs voice transfer from source to target. Hibiki also performs simultaneous S2ST and S2TT, while transferring voice characteristics. However, Hibiki relies on a decoder-only model which operates at a constant frame rate and performs inference with simple temperature sampling or greedy decoding. In particular, this allows for batching, unlike StreamSpeech's and Seamless's policies that involve a complex control flow that cannot be batched. This makes Hibiki able to translate hundreds of sequences on a single GPU, provides convenient support for classifier-free guidance, see e.g. Section 3.3, and allows to run in real time on device, as shown in Section 4.6.1. Human evaluations in Section 4.6 show that Hibiki significantly outperforms Seamless in terms of naturalness and audio quality, getting close to human interpretation.

2.3. Synthetic data for simultaneous translation

As pointed out by the IWSLT community (Ahmad et al., 2024), applying current state-of-the-art simultaneous translation technologies to complex scenarios (spontaneous speech, accents, background noise, dialogues, etc.) is still challenging. Unlike the offline text translation task, large interpretation datasets with voice preservation from source to target don't exist thus justifying the efforts to synthesize such data. Recent approaches (Wang et al., 2024) focused on speech-to-text and designed heuristics based on multilingual word aligners (Dyer et al., 2013) to create simultaneous translation examples from offline translation data. Similarly, the data processing behind the training dataset of Hibiki leverages the perplexity of an off-the-shelf translation system to build more robust word alignments designed for the streaming translation task. It also synthesizes target audios by maintaining high speech naturalness thus being the first implementation of a full synthetic data pipeline for simultaneous speech translation with voice preservation.

3. Method

We consider an utterance in a source language represented as a monophonic waveform $X \in \mathbb{R}^{f_s \cdot d}$, sampled at a frame rate $f_s = 24\,\text{kHz}$, of duration d. Similarly, its translation is given in a target language, denoted $Y \in \mathbb{R}^{f_s \cdot d}$. We assume X is padded to ensure both have the same duration. Our objective is to model $\mathbb{P}[Y|X]$. We further add the constraint that the modeling of Y knowing X should be causal and of minimal delay with respect to the source utterance, e.g. the same constraints that are imposed on a human interpreter in the context of live translation. To learn this constraint *via* supervised learning, Y must itself be built to respect this causality constraint. We first assume that Y respects this constraint, and we present how to model its distribution. Then, we introduce an information theory criterion to verify whether Y is causal with respect to X, and to adapt a non-causal interpretation into a causal one.

3.1. Modeling

We build on the framework introduced by Défossez et al. (2024) for the joint modeling of multiple sequences of discrete tokens, obtained from a neural audio codec.

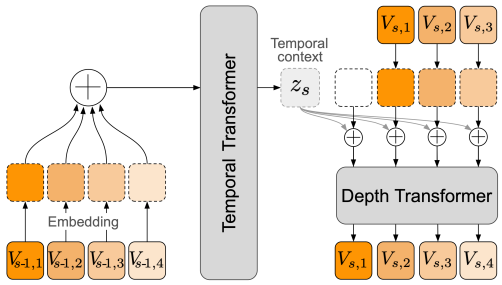

Figure 1. **Architecture of the RQ-Transformer.** With notations: $V_{t,q} = \tau(A)_{t,q}$. Figure adapted from Défossez et al. (2024).

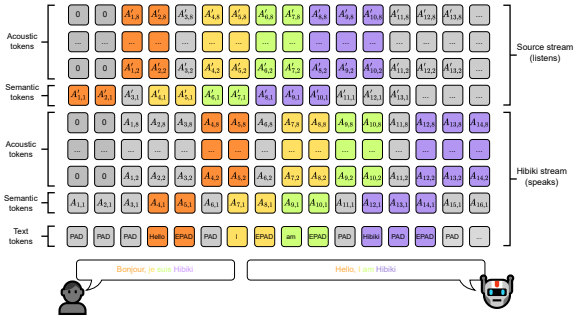

Figure 2. **Joint sequence modeling with contextual alignment.** From the source stream, Hibiki predicts its Inner Monologue text stream, and audio tokens. Its output is aligned for causality, as depicted in Figure 4. Figure adapted from Défossez et al. (2024).

3.1.1. Neural audio codec

We use the pre-trained causal and streaming Mimi codec (Défossez et al., 2024) to encode X and Y into low framerate sequences of discrete tokens. Mimi consists of an encoder and decoder from and to the waveform domain, and of an information bottleneck using Residual Vector Quantization (RVQ) (Zeghidour et al., 2022). The encoder transforms an input waveform of duration d into a latent vector $U \in \mathbb{R}^{C \times f_r \cdot d}$, with C the dimension of the latent space, and $f_r = 12.5\,\text{Hz}$ the frame rate. U is then projected to its nearest neighbor in a *codebook* table with N_A entries. The residual of the projection is further projected into a second table with the same cardinality, and so forth until Q projections have been performed. The last residual is discarded, and the decoder is trained to reconstruct the input waveform from the sum of the projected tensors. The codebooks are trained through exponential moving average, along with a commitment loss (Razavi et al., 2019). The rest of the model is trained only through an adversarial loss with feature matching (Défossez et al., 2024).

For language modeling, we are not interested in the quantized latent vector and its residuals, but in the discrete indices of the entry in the codebooks it is projected to. We denote those $(A_{t,q}) \in \{1, \dots, N_A\}^{f_r \cdot d \times Q}$. For Mimi we have $f_r = 12.5\,\text{Hz}$ and Q varies up to 32, but we use at most 16. Following Zhang et al. (2024b); Défossez et al. (2024), the output of the first quantization level is trained to replicate semantic information obtained from a WavLM self-supervised audio model (Chen et al., 2022). Following conventions of Borsos et al. (2022), we refer to $A_{t,1}$ as the *semantic* tokens, and $A_{t,q\geq 2}$ as the *acoustic* tokens. The acoustic tokens are arranged in a coarse to fine manner, the first ones have the most importance, and the latest model fine details of the audio and ensuring a smooth perception.

3.1.2. Joint modeling of discrete audio tokens

The discrete tokens for audio streams cannot easily be summarized into a single discrete sequence with reasonable cardinality and framerate (Copet et al., 2023). Following Yang et al. (2023); Défossez et al. (2024), we leverage a RQ-Transformer (Lee et al., 2022b) as shown in Figure 1 to model $(A_{t,q})$ both over the time t and quantizer q axes. It consists in a large *Temporal* Transformer (Vaswani et al., 2017), operating at the same framerate f_r as the codec, and being fed all the tokens generated so far, e.g. for all $t \leq f_r \cdot d$,

$$Z_t = \text{Temp}(A_0, \dots, A_{t-1}) \in \mathbb{R}^D. \tag{1}$$

A_0 is defined as a deterministic token indicating the start of the generation. Then, a smaller scale *Depth* Transformer models auto-regressively the tokens $A_{t,1}, \dots, A_{t,Q}$ over the quantizer axis, e.g. for all $t \leq f_r \cdot d$ and $q \leq Q$,

$$l_{t,q} = \text{Dep}(Z_t, A_{t,0}, \dots, A_{t,q-1}) \in \mathbb{R}^{N_a}, \tag{2}$$

with $A_{t,0}$ also a special token, and with the goal of having,

$$\text{softmax}(l_{t,q}) \approx \mathbb{P}\left[A_{t,q} | A_0, \dots, A_{t-1}, A_{t,0}, \dots A_{t,q-1}\right]$$

Following (Copet et al., 2023; Défossez et al., 2024), we further introduce an acoustic delay of 2 time steps, meaning that we model $(\tau(A)_{t,q})$ instead of A,

$$\begin{cases} \tau(A)_{t,1} &= A_{t,1} & \forall\, t \\ \tau(A)_{t,q} &= A_{t-2,q} & \forall\, t \geq 3, \forall\, q \geq 2 \\ \tau(A)_{t,q} &= 0 & \forall\, t < 3, \forall\, q \geq 2, \end{cases} \tag{3}$$

with 0 being a special token. The delay is removed before decoding audio with the codec.

3.1.3. Translation as multistream modeling

We have presented how the RQ-Transformer given by eq. (1) and (2) allows for jointly modeling multiple discrete streams of tokens. We adapt this framework for the task

of joint speech-to-speech and speech-to-text simultaneous translation as illustrated in Figure 2. We concatenate the audio tokens A^Y obtained from the target interpretation Y, with the tokens A^X from the source utterance X along the q-axis, e.g.

$$\bar{A} = \text{concat}_\text{q}\left[\tau(A^Y), \tau(A^X)\right]. \quad (4)$$

We observe a benefit from modeling the tokens A^X at train time, although at inference time, predictions for those tokens are skipped and actual tokens of the input are used instead.

Défossez et al. (2024) showed generating an Inner Monologue, i.e. padded text tokens aligned with the content of the generated audio, is beneficial to the quality and stability of the generated audio. This is similar to multi-task learning where the translation is predicted both in the audio and text domain. Hibiki thus also predicts a text stream corresponding to the transcription of the output Y, with sufficient padding between words to keep them aligned with the audio. Note that unlike previous multi-task translation work, Hibiki makes active use of this capability at inference time. We denote W_t the text stream, with cardinality N_W and the same frame rate f_r as the audio streams.

3.1.4. Architectural details

We provide architectural hyper-parameters in Section 4.1. At time-step t, the tokens from the step $t-1$, e.g. $\tau(A^X)_{t-1}$, $\tau(A^Y)_{t-1}$, and W_{t-1}, are fed into dedicated embedding tables, whose contributions are summed. For the first time step $t = 1$, a BOS token is used instead. We then use standard Transformer layers (Vaswani et al., 2017), with gated SiLU activation (Shazeer, 2020; Hendrycks & Gimpel, 2016). A linear layer maps its output Z_t to logits for the text token W_t. The *Depth* Transformer then operates for $2 \cdot Q$ steps: the first half to estimate the logits for the output stream, and the next half for the input stream. Each depth step q takes as input Z_t summed with a learnt embedding of the previous audio token $\bar{A}_{t,q-1}$, or W_t for $q = 1$.

3.2. Alignment and synthetic interpretation data

We have assumed pairs (X, Y) that respect the constraint of simultaneous interpretation. We now introduce an unsupervised criterion to estimate and enforce causality dependencies between the source and target utterances.

3.2.1. Text domain alignments

Let us first express formally those constrained in the text domain. Let us take $S = (S_1, \ldots, S_n)$ the sequence of words in the utterance X, and $T = (T_1, \ldots, T_m)$ that in Y.

Ideal alignment. We seek to define an ideal alignment $(a_j^{\text{ideal}}) \in \{1, \ldots, n\}^m$, where a_j^{ideal} indicates the index of the word in S that the j-th word in T should wait for to minimize the uncertainty on T_j. Any alignment strictly less conservative than a^{ideal} would risk the model hallucinating at inference if trained on. Any alignment strictly more conservative would still be causal, but introduce more latency.

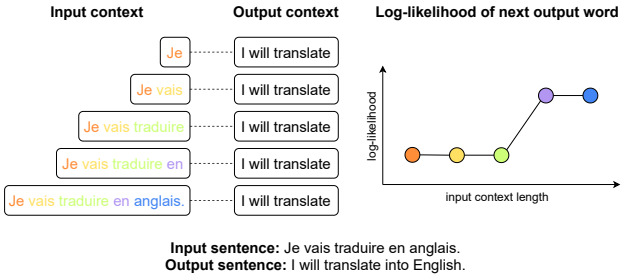

Figure 3. **Contextual alignment.** We compute the log-likelihood of the word "into" with a pre-trained text translation model, for various input truncations. Once the matching source word "en" appears, we observe a large increase in log-likelihood, see eq. (6).

Contextual alignment. We introduce a criterion to estimate a^{ideal}. Let's denote the conditional log-likelihood

$$\log(p_{j,i}) = \log\left(\mathbb{P}\left[T_j | S_1, \ldots S_i, T_1, \ldots, T_{j-1}\right]\right), \quad (5)$$

we expect $\log p_{j,i}$ to increase with i, as more context is beneficial. We conjecture that $\log(p_{j,i}) - \log(p_{j,i-1})$ is maximal for $i = a_j$. We compute an estimate $\log(\hat{p}_{j,i})$ of $\log(p_{j,i})$ with an off-the-shelf text translation language model MADLAD-3B (Kudugunta et al., 2023), by feeding it truncated input up to the i-th word, which we use to define a *contextual* alignment, illustrated in Figure 3,

$$a_j^{\text{ctx}} = \arg\max_{i \leq n} \left[\log(\hat{p}_{j,i}) - \log(\hat{p}_{j,i-1})\right]. \quad (6)$$

Examples of alignments are given in the Appendix, Figure 9.

3.2.2. Audio domain alignments

Given (X, Y), we transcribe both with timestamps with a Whisper model (Radford et al., 2023; Louradour, 2023) and apply eq. (6). The pair (X, Y) respects the alignment (a_j^{ctx}) if the timestamp of the j-th word in Y comes after the timestamp of the a_j-th word in X. To reduce the impact of errors, we require Y to lag by at least $\delta^{\text{ctx}} = 2$ sec. compared to the contextual alignment, and cut spikes higher than 25% of the average delay over a window of 5 words.

Silence insertion. If Y doesn't respect the alignment, one can simply transform it by inserting sufficient silences before a word, as illustrated in Figure 4, with two limitations:
(i) silence insertion can lead to hard cuts when the timestamps are inaccurate or no pause exists between words;
(ii) the corrected Y might be arbitrarily late on the ideal alignment, e.g. if the speech rate is slower in Y than in X.
We apply this method during the speech translation training.

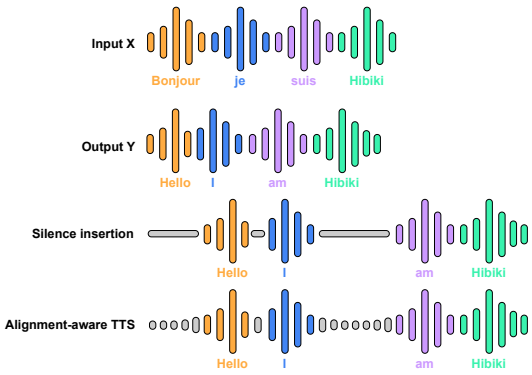

Figure 4. **Generating aligned interpretations.** We extract unsupervised word level contextual alignment, which we lift to audio by either inserting silences, or re-synthesising with an alignment aware TTS. See Section 3.2 for details.

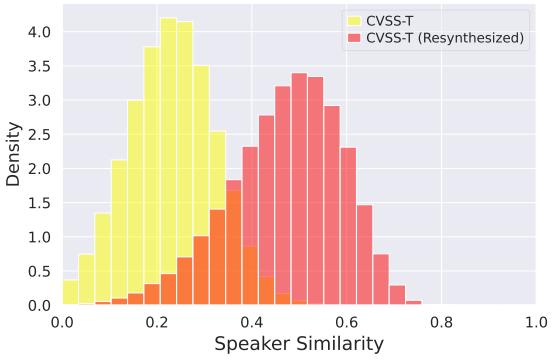

Figure 5. Speaker similarity between source and target speech in CVSS-T training data, before and after resynthesis.

Alignment-aware TTS. We obtain more natural aligned data by (re)-synthesising Y with a TTS model able to follow hard and soft constraints on word locations, along with accurate speaker conditioning. For existing datasets, this can have the added benefit of improving the word error rate and the speaker similarity, as illustrated in Section 3.3. Following (Défossez et al., 2024), Appendix C, we train a TTS with both audio and a synced text stream as output, along with voice conditioning. The text stream is constrained to match exactly the text to generate, with the model having only the freedom to insert padding tokens. The audio output is late on the text, so that its content is conditioned by it, both for content and timestamps. If the TTS is early on the alignment a^{ctx}, padding tokens are forced to delay the next word. When the TTS is lagging on its target, a penalty is added on the logits of the padding token. The penalty scales from 0 to -2 as the lag increases from 1 to 2 seconds. This increases smoothly the rate of speech to catch up with the source audio. We perform 6 to 8 generations per input, and select the best one based on word error rate first, and speaker similarity second. We apply this only for the fine-tuning speech translation dataset.

3.3. Voice Transfer

Improving voice transfer data. Training speech translation models with voice transfer typically amounts to supervised training on synthetic paired sequences of the same speaker. In particular, CVSS-T (Jia et al., 2022b) is the standard training set for S2ST with voice transfer and provides such artificial targets. However, Figure 5 shows that the average speaker similarity—as measured by the cosine similarity between speaker embeddings of source and target— on this dataset is very low with an average of 0.23. As a calibration, state-of-the-art cross-lingual voice transfer lies around 0.40 (Rubenstein et al., 2023). We thus also regenerate CVSS-T with our alignment-aware TTS, as it allows for voice transfer. As shown in Figure 5, the resynthesized CVSS-T displays a higher similarity, with an average of 0.47. Yet, our training mixture which combines synthetic data and resynthesized CVSS-T still covers a wide range, with a significant mass below 0.40.

Conditional training. Filtering training data to only keep pairs of examples with a high similarity would improve voice transfer, but the resulting reduction in training data would likely affect translation quality (e.g. keeping only samples with a speaker similarity above 0.40 would remove 45% of training data). We rather rely on conditional training (Keskar et al., 2019; Korbak et al., 2023) to inform the generative model of how reliable each training example is in terms of voice transfer. We label each training sample with a discrete "voice transfer score" in $\{\texttt{very\_bad}, \texttt{bad}, \texttt{neutral}, \texttt{good}, \texttt{very\_good}\}$ based on quantiles of speaker similarity, each label being associated to a learnable embedding added to the model's inputs at every timestep. Importantly, the quantiles are computed *before* combining the synthetic data and CVSS-T to guarantee that the model does not associate a specific label to a specific dataset rather than the actual speaker similarity. At inference time, we always pass the `very_good` label.

Classifier-free guidance. Following Kreuk et al. (2023) we can increase the impact of the conditioning by using classifier-free guidance. We compute logits both with conditionings `very_good` and `very_bad` and sample from:

$$\gamma l_{t,q}^{\texttt{very\_good}} + (1-\gamma) l_{t,q}^{\texttt{very\_bad}}, \tag{7}$$

which is compatible with real-time inference by producing both sets of logits with a batch size of 2. Section 4.6 shows that it significantly improves voice transfer.

4. Experiments

4.1. Architectural hyper-parameters

Hibiki consists of a *Temporal* Transformer with a latent dimension of 2560 (7040 for the SiLU gating), 24 layers, 20 heads and local attention over 1500 tokens, i.e. 2.2B parameters and a 120s context. The *Depth* Transformer initially follows Défossez et al. (2024), i.e. 6 layers per codebook, a latent dimension of 1024 (2816 for the gating), and 16 heads. It models $Q = 16$ audio codebooks for the output stream, and the same for the input stream (only at training). To reduce the footprint of the *Depth* Transformer we distill it post-training into a smaller one, with 4 layers per codebook, weight sharing for codebooks 9 to 16, low-rank embedding tables of dimension 128, and a dimension of 2048 for the gating. This reduces its size from 1.1B parameters to 449M parameters, for a total of 2.7B parameters. We also train Hibiki-M as a distilled version of Hibiki, a 1.8B variant capable of running in real-time on device, using a *Temporal* Transformer with a latent dimension of 2048, 16 layers, and only 8 codebooks levels per stream. Architectural hyper-parameters are summarized in Table 6.

4.2. Training protocol

We train a French-English speech translation system through the following steps, each with a cosine learning rate schedule and AdamW (Loshchilov & Hutter, 2019), with a weight decay of 0.1, and momentum parameters of (0.9, 0.95).

Text pretraining. We first pretrain the *Temporal* Transformer from scratch on multilingual text-only data using next token prediction, for 600K steps, with a batch of 1,024 sequences of length 4,096. We use a cosine learning rate schedule, with 2K warmup steps and a maximum value of $4.8 \cdot 10^{-4}$. Our training dataset is made of filtered web pages from Common Crawl, as well as curated sources such as Wikipedia, StackExchange or scientific articles and it contains 12.5% of multilingual documents.

Audio pretraining. Starting from the pretrained text model, we perform an audio pretraining on non-parallel French and English data with a single stream as done by Défossez et al. (2024). We train for 1,450K steps with a batch size of 144 and a learning rate of $2 \cdot 10^{-4}$. Then we duplicate the weights of the *Depth* Transformer for multistream modeling.

Speech translation training. We build a French-English speech translation dataset of approximately 40K hours in each language. Starting from a collection of expressive audio content in French, we extract roughly 2.3M single speaker utterances each with a duration around 60 seconds. We transcribe these segments with Whisper (Radford et al., 2023), using the large-v3 model. We rely on PySBD (Sadvilkar & Neumann, 2020) to segment each transcript into sentences and use MADLAD-3B (Kudugunta et al., 2023) to translate them individually, before joining them back into a translated English transcript. We synthesize each with the TTS described in Section 3.2.2, conditioned on the original French speaker identity with a 10s utterance. We apply the silence insertion technique described in Section 3.2.2 to obtain simultaneous interpretation pairs. We train for 150K steps with a batch size of 96, a learning rate of $3 \cdot 10^{-5}$ and compute the loss on both the source and the target streams. We use conditional training on speaker similarity as explained in 3.3 and apply noise augmentation techniques on the source audio. For each training pair, we introduce a special input EOS token on all audio tokens of the source at the first frame after the end of its speech utterance and use another special EOS token on the text tokens stream to indicate the end of the model speech utterance.

Speech translation fine-tuning. We use alignment-aware TTS generations introduced in Section 3.2.2 to build a synthetic dataset composed of long-form utterances and an improved version of CVSS-T/train, with natural pauses and high speaker similarity, totaling close to 900 hours. We fine-tune for 8K steps with a batch size of 8, a learning rate of $2 \cdot 10^{-6}$, conditional training on the speaker similarity, special EOS tokens, and apply the loss to both streams.

Training of Hibiki-M. It goes through the same text and audio pre-training stages. During the speech translation training it is soft distilled from Hibiki, before going through the same fine-tuning stage without distillation.

Table 1. **Comparison with offline baselines.** We also report performance from a closed-source streaming model (*) as it uses the same evaluation protocol.

MODEL	ASR-BLEU
TRANSLATOTRON (JIA ET AL., 2019)	17.0
TRANSLATOTRON 2 (JIA ET AL., 2022A)	26.0
S2UT (LEE ET AL., 2022A)	22.2
UNITY (INAGUMA ET AL., 2023)	27.8
DASPEECH (FANG ET AL., 2023)	25.0
RNN-TRANSDUCER* (ZHAO ET AL., 2024)	25.4
STREAMSPEECH (OFFLINE) (ZHANG ET AL., 2024A)	28.5
HIBIKI	**30.5**

4.3. Evaluation metrics and baselines

We first compare Hibiki to several offline baselines as well as a closed source streaming baseline (Zhao et al., 2024). We then perform a comparison between Hibiki and the two existing methods for simultaneous translation: Seamless (Barrault et al., 2023) and StreamSpeech (Zhang et al., 2024a) with a chunk size of 2560ms. We evaluate translation and audio quality, naturalness, speaker similarity and translation latency using automatic and human evaluations.

Table 2. Objective comparison of Hibiki with StreamSpeech (Zhang et al., 2024a) and Seamless (Barrault et al., 2023).

	Short-form (CVSS-C Fr-En test)						Long-form (Audio-NTREX)					
Model	BLEU (↑)	ASR BLEU (↑)	ASR COMET (↑)	Speaker Sim. (↑)	End Offset (↓)	LAAL (↓)	BLEU (↑)	ASR BLEU (↑)	ASR COMET (↑)	Speaker Sim. (↑)	End Offset (↓)	LAAL (↓)
MADLAD-3B	-	-	94.2	-	-	-	-	-	72.6	-	-	-
StreamSpeech	26.4	25.4	67.0	-	1.6	2.8	0.1	0.1	18.4	-	N/A	N/A
Seamless	37.0	33.8	77.6	0.30	**1.4**	**2.8**	25.4	23.9	27.3	0.43	**1.6**	**4.2**
Hibiki-M	37.5	33.7	75.7	0.34	2.8	3.5	25.9	25.0	30.5	0.39	2.3	5.5
Hibiki	**39.2**	**35.5**	**78.3**	**0.41**	2.9	3.4	**27.5**	**26.6**	**34.9**	**0.48**	2.7	5.0

Translation quality. We evaluate translation quality by transcribing generated speech and computing BLEU (Post, 2018) and COMET (Rei et al., 2020) scores with the reference, referred to as ASR-BLEU and ASR-COMET. When comparing to offline baselines in Table 1, we replicate the setting of Zhang et al. (2024a) for a fair comparison. As we observed frequent ASR mistakes, we rather used Whisper medium (Radford et al., 2023) in subsequent experiments and compute BLEU/ASR-BLEU scores after hypothesis and ground-truth text normalization.[3] Since Seamless, StreamSpeech and Hibiki also produce a text translation, we also evaluate their BLEU score using the same text normalization. ASR-COMET scores are calculated without text normalization and using the XCOMET-XL model.[4]

Audio quality and naturalness. Human raters evaluate the audio quality of generated speech and its naturalness. We evaluate the latter as a proxy for "realism": are the flow and prosody natural, are pauses smooth and properly placed or are there abrupt cuts? We compute each score for each model by averaging Mean Opinion Scores between 1 and 5 across 30 samples, each sample being evaluated by 15 raters.

Cross-lingual speaker similarity. For objective evaluation, we use a standard model for speaker verification[5] based on WavLM (Chen et al., 2022) and report the cosine similarity between the embeddings of the source and the generated speech. To mitigate potential biases due to using the same speaker verification model for conditional training (see Section 3.3), we also collect human judgments where raters are asked to rate the similarity to the source audio.

Latency. A metric for S2ST latency is the End Offset, which is the time (in seconds) between the end of the last word of the source and that of the last word in the output. We also measure the Length-Adaptative Average Lagging (LAAL) following the method described by Papi et al. (2022): it approximates the average time (in seconds) between the pronunciation of a source word and its translation, without requiring word-level alignments. We rely on word-level emission timestamps $(d_i)_{1 \dots n_{\text{gen}}}$ produced by Whisper for n_{gen} words in the generated speech. We define $\epsilon = \frac{\Delta_{\text{source}}}{max(n_{\text{gen}}, n_{\text{ref}})}$ where Δ_{source} is the duration of the source speech and n_{ref} the number of words in the reference translation. The LAAL score is then computed as $\frac{1}{n_{\max}} \sum_{i=1}^{n_{\max}} d_i - (i-1)\epsilon$ where $n_{max} = \min\{i | d_i \geq \Delta_{\text{source}}\}$.

[3] github.com/openai/whisper/blob/main/whisper/normalizers
[4] github.com/Unbabel/COMET
[5] github.com/microsoft/UniSpeech ("WavLM Large")

4.4. Evaluation datasets

Short-form data. We evaluate models on the Fr-En task of CVSS (Jia et al., 2022b). While it is the standard benchmark for S2ST and allows comparisons with previous models, we observe that 99% of its sequences are shorter than 10 seconds. We thus extend our evaluation to long-forms.

Long-form data. We collect long-form speech translations by recording bilingual speakers as they read Fr-En translations from the NTREX (Aepli et al., 2023) text corpus. This speech corpus, that we name Audio-NTREX contains 10 hours of real human speech in each language with 10 different speakers and an average of 50 sec. per utterance.

Real interpretation. To compare with human interpreters, we use 90 real interpretations of the European Parliament from VoxPopuli (Wang et al., 2021a) where translations contain the source speech at a lower volume. For a fair comparison, we also add the lowered source speech to generations of Hibiki and Seamless (see our external webpage for samples).

4.5. Inference configuration

We encode audio with the streaming codec and feed the tokens to Hibiki while decoding the output tokens to obtain streaming translation. At the end of the input, we send the EOS token to our model, and keep sampling until it produces its own EOS. Inference parameters are cross validated independently for each dataset using a held-out 8% of Audio-NTREX and the valid split of CVSS-C. The optimal parameters are $\gamma = 3.0$, a temperature of 0.8, top-k of 250 for audio tokens and 50 for text tokens for Audio-NTREX. On CVSS, the same configuration is used except for text tokens that are sampled with a temperature of 0.1. We conjecture that the lower text temperature typically improves translation but can lead to producing an EOS token too early.

Table 3. **Human evaluation.** Raters report Mean Opinion Scores (MOS) between 1 and 5. Ground-truth is real human interpretation.

MODEL	QUALITY	SPEAKER SIM.	NATURALNESS
GROUND-TRUTH	4.18 ± 0.07	-	4.12 ± 0.08
SEAMLESS	2.22 ± 0.08	2.86 ± 0.12	2.18 ± 0.09
HIBIKI	$\mathbf{3.78} \pm 0.09$	$\mathbf{3.43} \pm 0.10$	$\mathbf{3.73} \pm 0.09$

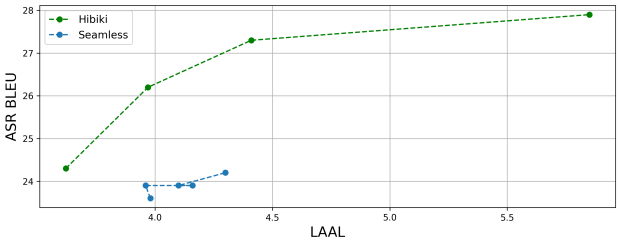

Figure 6. Quality/Latency trade-off of multiple Hibiki models vs. Seamless with different decision thresholds on Audio-NTREX.

4.6. Results

Translation quality. Table 1 compares Hibiki with offline baselines that have access to the complete source audio when translating. Despite performing simultaneous translation, Hibiki outperforms all models, including the offline variant of StreamSpeech. Table 2 benchmarks Hibiki against available baselines for simultaneous translation. In the short-form setting, our model outperforms StreamSpeech and Seamless at the cost of an average 0.7s of additional lagging. The long-form dataset represents a more significant challenge, as StreamSpeech does not manage to produce intelligible translations. Hibiki outperforms Seamless, again with a latency higher by an average of 0.8s.

Audio fidelity. Objective evaluations for speaker similarity, as reported in Table 2, show that Hibiki demonstrates significantly better voice transfer than Seamless (we do not evaluate StreamSpeech as it does not perform voice transfer). Human evaluations reported in Table 3 confirm this result and furthermore show a much higher quality and naturalness than Seamless, that get close to that of ground-truth audio from professional human interpreters. This means that Hibiki not only produces high-quality audio, but it inserts smooth and natural pauses into its flow.

Quality/Latency trade-off. The quality/latency trade-off of Hibiki cannot be controlled at inference as its translation policy is conditioned by the data preparation process described in Section 3.2.2. However, by varying the lag δ^{ctx} added between contextually aligned words, one can set the quality/latency trade-off at training time. We compared different versions of Hibiki trained with δ^{ctx} of 0.5, 1, 1.5 and 3 sec. against Seamless parameterized with different decision threshold values from 0.3 to 0.7. Figure 6 shows that Hibiki achieves a better trade-off than Seamless and allows to design more steerable systems.

Preliminary English-to-French results. Using the silence insertion technique, we also built a English-to-French speech translation dataset and trained another Hibiki model whose objective evaluation results are given in Table 7.

Ablation: alignment strategies. We compare our contextual alignment method with alternatives. Table 4 shows that applying no lag to the target speech during training results in very low translation quality, which can be expected since the model lacks context to produce the translation. Adding lag in training examples improves the ASR-BLEU, with 10 seconds representing a reasonable value, however the resulting average latency (as represented by LAAL) is much worse than using a contextual alignment, as the model does not adapt its flow to the context. A middle ground between constant lag and contextual alignment is that of "sentence alignment" that simply moves the start of each output sentence to the end of the corresponding source sentence. This improves translation quality, however degrading the latency even more. Overall, contextual alignment provides the best trade-off between translation quality and latency.

Table 4. Ablations.

MODEL	ASR-BLEU	LAAL
NO LAG	4.2	2.46
CONSTANT LAG (2S)	10.0	2.49
CONSTANT LAG (10S)	22.5	9.02
SENTENCE ALIGNMENT	25.6	21.49
NO INNER MONOLOGUE	17.1	14.34
NO AUDIO PRETRAINING	14.6	5.12
HIBIKI	26.6	5.0

Ablation: Classifier-free guidance. Table 5 shows that using the `very_good` label provides a speaker similarity of 0.42, similar to that of Seamless (0.43). Using classifier-free guidance with $\gamma = 3.0$ significantly improves it without significantly hurting translation quality, while increasing its weight too much results in degraded performance due to unintelligible speech. Supplementary material interestingly illustrates how increasing γ to extreme values results in an exaggerated French accent (the source language in our experiments), which we can attribute to biases in the speaker model used to label our data.

Ablation: General ablations. Section 3.1.3 describes how jointly predicting text tokens serves as a scaffolding for audio generation. Table 4 illustrates this claim as training Hibiki in a unimodal fashion, without predicting text outputs, results in much worse performance, as does starting from a pretrained text LM and training directly for S2ST.

Table 5. Ablations on classifier-free guidance.

CFG PARAM.	ASR-BLEU	SPEAKER SIM.
NO CFG	26.0	0.42
$\gamma = 3.0$ (DEFAULT)	26.6	0.48
$\gamma = 10.0$	18.9	0.44

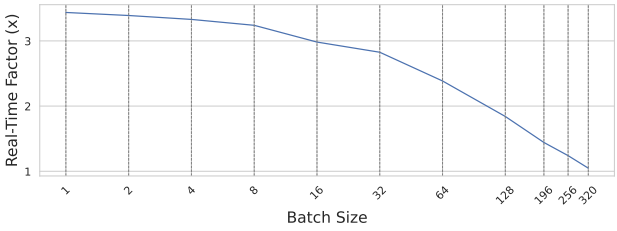

Figure 7. Batched inference speed of Hibiki on a H100 SXM.

4.6.1. INFERENCE CAPABILITIES

Batched inference. Hibiki's inference uses temperature sampling with a constant framerate. This makes it easy to perform streaming classifier-free guidance as well as batching the processing of several sources of speech at the same time. This is unlike Seamless and StreamSpeech whose complex inference policies are difficult to batch as they require dynamic, irregular decisions for each sequence. Figure 7 shows that Hibiki remains faster than real-time on a H100 even when processing 320 sequences in parallel (or 160 with classifier-free guidance).

On-device inference. Our distilled Hibiki-M is competitive with Seamless on short-form and long-form translation as shown in Table 2. We attribute the lower speaker similarity on long-form audio to the lower number of quantizers modeled by Hibiki-M (8 instead of 16) which results in a twice lower audio bitrate. Figure 8 shows inference traces of Hibiki-M on an iPhone 16 Pro. Hibiki-M remains faster than real-time along a minute of inference, even with a batch size of 2 which is necessary for classifier-free guidance. Training Hibiki-M with sliding window attention would furthermore improve real-time factor along time.

4.6.2. LIMITATIONS

This study focuses on a single translation task (French to English) and scaling to more languages could benefit from MADLAD which is massively multilingual, however it would require training TTS systems on more languages. Moreover, while Hibiki reaches 35.5 ASR-BLEU against CVSS-C ground-truth targets, it reaches 47.9 ASR-BLEU if compared to MADLAD text translations instead. This shows that Hibiki is excellent at predicting translations that could be produced by MADLAD, and training it to predict pseudo-targets from better or more diverse translations can improve translation quality w.r.t ground-truth targets.

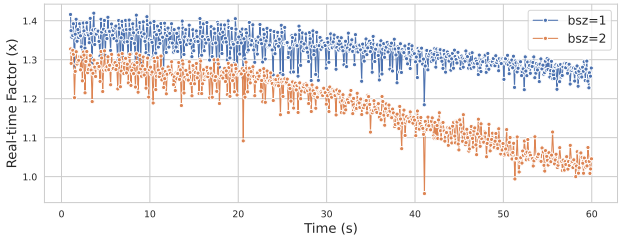

Figure 8. Inference speed of Hibiki-M on an iPhone 16 Pro.

5. Conclusion

We introduce Hibiki, a model for simultaneous and expressive speech and text translation. Hibiki leverages a multistream architecture that casts live translation as simple temperature sampling, thanks to a weakly-supervised method for aligning paired translation data. Hibiki is competitive with the state-of-the-art in terms of translation quality, while demonstrating a much more natural flow close to human interpretation as well as a better voice transfer. Moreover, Hibiki is compatible with streaming batched inference, which facilitates large-scale deployment, while the smaller Hibiki-M runs in real-time on a smartphone.

Impact Statement

Our work aims at bringing simultaneous speech translation closer to real-world applications, both by improving its quality and by making it deployable on-device. This carries a potential for improving communication between humans. We acknowledge that speech translation may affect employment opportunities for interpreters in contexts such as international meetings or live interviews, however we believe that it will will have a net positive impact in all contexts where an interpreter would not be considered, e.g. when traveling or streaming videos. We also acknowledge risks associated to voice transfer, while we remark that S2ST with voice transfer represents a lower risk of spoofing than TTS as the output textual content is not directly controllable and the model is only given a limited freedom to reformulate its inputs.

References

Aepli, N., Amrhein, C., Schottmann, F., and Sennrich, R. A benchmark for evaluating machine translation metrics on dialects without standard orthography. In Koehn, P., Haddon, B., Kocmi, T., and Monz, C. (eds.), *Proceedings of the Eighth Conference on Machine Translation, WMT 2023*, pp. 1045–1065. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.WMT-1.99.

Ahmad, I. S., Anastasopoulos, A., Bojar, O., Borg, C., Carpuat, M., Cattoni, R., Cettolo, M., Chen, W., Dong, Q., Federico, M., Haddow, B., Javorský, D., Krubin-

ski, M., Lam, T. K., Ma, X., Mathur, P., Matusov, E., Maurya, C., McCrae, J. P., Murray, K., Nakamura, S., Negri, M., Niehues, J., Niu, X., Ojha, A. K., Ortega, J. E., Papi, S., Polák, P., Pospísil, A., Pecina, P., Salesky, E., Sethiya, N., Sarkar, B., Shi, J., Sikasote, C., Sperber, M., Stüker, S., Sudoh, K., Thompson, B., Turchi, M., Waibel, A., Watanabe, S., Wilken, P., Zemánek, P., and Zevallos, R. Findings of the IWSLT 2024 evaluation campaign. *CoRR*, abs/2411.05088, 2024. doi: 10.48550/ARXIV.2411.05088. URL https://doi.org/10.48550/arXiv.2411.05088.

Barrault, L., Chung, Y., Meglioli, M. C., Dale, D., Dong, N., Duppenthaler, M., Duquenne, P., Ellis, B., Elsahar, H., Haaheim, J., Hoffman, J., Hwang, M., Inaguma, H., Klaiber, C., Kulikov, I., Li, P., Licht, D., Maillard, J., Mavlyutov, R., Rakotoarison, A., Sadagopan, K. R., Ramakrishnan, A., Tran, T., Wenzek, G., Yang, Y., Ye, E., Evtimov, I., Fernandez, P., Gao, C., Hansanti, P., Kalbassi, E., Kallet, A., Kozhevnikov, A., Gonzalez, G. M., Roman, R. S., Touret, C., Wong, C., Wood, C., Yu, B., Andrews, P., Balioglu, C., Chen, P., Costa-jussà, M. R., Elbayad, M., Gong, H., Guzmán, F., Heffernan, K., Jain, S., Kao, J., Lee, A., Ma, X., Mourachko, A., Peloquin, B. N., Pino, J., Popuri, S., Ropers, C., Saleem, S., Schwenk, H., Sun, A. Y., Tomasello, P., Wang, C., Wang, J., Wang, S., and Williamson, M. Seamless: Multilingual expressive and streaming speech translation. *CoRR*, abs/2312.05187, 2023. doi: 10.48550/ARXIV.2312.05187.

Berard, A., Pietquin, O., Servan, C., and Besacier, L. Listen and translate: A proof of concept for end-to-end speech-to-text translation. *CoRR*, abs/1612.01744, 2016.

Borsos, Z., Marinier, R., Vincent, D., Kharitonov, E., Pietquin, O., Sharifi, M., Roblek, D., Teboul, O., Grangier, D., Tagliasacchi, M., and Zeghidour, N. Audiolm: A language modeling approach to audio generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2022.

Chen, S., Wang, C., Chen, Z., Wu, Y., Liu, S., Chen, Z., Li, J., Kanda, N., Yoshioka, T., Xiao, X., Wu, J., Zhou, L., Ren, S., Qian, Y., Qian, Y., Wu, J., Zeng, M., Yu, X., and Wei, F. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE J. Sel. Top. Signal Process.*, 2022.

Copet, J., Kreuk, F., Gat, I., Remez, T., Kant, D., Synnaeve, G., Adi, Y., and Défossez, A. Simple and controllable music generation. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023*, 2023.

Défossez, A., Mazaré, L., Orsini, M., Royer, A., Pérez, P., Jégou, H., Grave, E., and Zeghidour, N. Moshi: a speech-text foundation model for real-time dialogue. *CoRR*, abs/2410.00037, 2024. doi: 10.48550/ARXIV.2410.00037.

Dyer, C., Chahuneau, V., and Smith, N. A. A simple, fast, and effective reparameterization of IBM model 2. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013*, pp. 644–648, 2013. URL https://aclanthology.org/N13-1073/.

Fang, Q., Zhou, Y., and Feng, Y. Daspeech: Directed acyclic transformer for fast and high-quality speech-to-speech translation. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023*, 2023.

Hendrycks, D. and Gimpel, K. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Inaguma, H., Popuri, S., Kulikov, I., Chen, P.-J., Wang, C., Chung, Y.-A., Tang, Y., Lee, A., Watanabe, S., and Pino, J. UnitY: Two-pass direct speech-to-speech translation with discrete units. In Rogers, A., Boyd-Graber, J., and Okazaki, N. (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15655–15680, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.872.

Jain, A. N., McNair, A. E., Waibel, A., Saito, H., Hauptmann, A., and Tebelskis, J. Connectionist and symbolic processing in speech-to-speech translation: The JANUS system. In *Proceedings of Machine Translation Summit III: Papers*, pp. 113–117, Washington DC, USA, July 1-4 1991.

Jia, Y., Weiss, R. J., Biadsy, F., Macherey, W., Johnson, M., Chen, Z., and Wu, Y. Direct Speech-to-Speech Translation with a Sequence-to-Sequence Model. In *Proc. Interspeech 2019*, pp. 1123–1127, 2019. doi: 10.21437/Interspeech.2019-1951.

Jia, Y., Ramanovich, M. T., Remez, T., and Pomerantz, R. Translatotron 2: High-quality direct speech-to-speech translation with voice preservation. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 10120–10134. PMLR, 17–23 Jul 2022a.

Jia, Y., Ramanovich, M. T., Wang, Q., and Zen, H. CVSS corpus and massively multilingual speech-to-speech translation. In Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Odijk, J., and Piperidis, S. (eds.), *Proceedings of the Thirteenth Language Resources and Evaluation Conference, LREC 2022*, pp. 6691–6703. European Language Resources Association, 2022b.

Keskar, N. S., McCann, B., Varshney, L. R., Xiong, C., and Socher, R. CTRL: A conditional transformer language model for controllable generation. *CoRR*, abs/1909.05858, 2019.

Korbak, T., Shi, K., Chen, A., Bhalerao, R. V., Buckley, C. L., Phang, J., Bowman, S. R., and Perez, E. Pretraining language models with human preferences. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023*, volume 202 of *Proceedings of Machine Learning Research*, pp. 17506–17533. PMLR, 2023.

Kreuk, F., Synnaeve, G., Polyak, A., Singer, U., Défossez, A., Copet, J., Parikh, D., Taigman, Y., and Adi, Y. Audiogen: Textually guided audio generation. In *The Eleventh International Conference on Learning Representations, ICLR 2023*. OpenReview.net, 2023.

Kudugunta, S., Caswell, I., Zhang, B., Garcia, X., Xin, D., Kusupati, A., Stella, R., Bapna, A., and Firat, O. MADLAD-400: A multilingual and document-level large audited dataset. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023*, 2023.

Lee, A., Chen, P.-J., Wang, C., Gu, J., Popuri, S., Ma, X., Polyak, A., Adi, Y., He, Q., Tang, Y., Pino, J., and Hsu, W.-N. Direct speech-to-speech translation with discrete units. In Muresan, S., Nakov, P., and Villavicencio, A. (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3327–3339, Dublin, Ireland, May 2022a. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.235.

Lee, D., Kim, C., Kim, S., Cho, M., and Han, W. Autoregressive image generation using residual quantization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 11513–11522. IEEE, 2022b. doi: 10.1109/CVPR52688.2022.01123.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019*, 2019.

Louradour, J. whisper-timestamped. https://github.com/linto-ai/whisper-timestamped, 2023.

Ma, X., Wang, Y., Dousti, M. J., Koehn, P., and Pino, J. M. Streaming simultaneous speech translation with augmented memory transformer. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2021*, pp. 7523–7527. IEEE, 2021. doi: 10.1109/ICASSP39728.2021.9414897.

Nakamura, S., Markov, K., Nakaiwa, H., Kikui, G., Kawai, H., Jitsuhiro, T., Zhang, J.-S., Yamamoto, H., Sumita, E., and Yamamoto, S. The ATR multilingual speech-to-speech translation system. *IEEE Transactions on Audio, Speech, and Language Processing*, 2006.

Papi, S., Gaido, M., Negri, M., and Turchi, M. Over-generation cannot be rewarded: Length-adaptive average lagging for simultaneous speech translation. *CoRR*, abs/2206.05807, 2022. doi: 10.48550/ARXIV.2206.05807.

Post, M. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pp. 186–191, Brussels, Belgium, October 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-6319.

Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., and Sutskever, I. Robust speech recognition via large-scale weak supervision. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *International Conference on Machine Learning, ICML 2023*, volume 202 of *Proceedings of Machine Learning Research*, pp. 28492–28518. PMLR, 2023.

Razavi, A., van den Oord, A., and Vinyals, O. Generating diverse high-fidelity images with vq-vae-2. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Rei, R., Stewart, C., Farinha, A. C., and Lavie, A. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020*, pp. 2685–2702, 2020. doi: 10.18653/V1/2020.EMNLP-MAIN.213. URL https://doi.org/10.18653/v1/2020.emnlp-main.213.

Ren, Y., Liu, J., Tan, X., Zhang, C., Qin, T., Zhao, Z., and Liu, T.-Y. SimulSpeech: End-to-end simultaneous speech to text translation. In Jurafsky, D., Chai,

J., Schluter, N., and Tetreault, J. (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 3787–3796, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.350.

Rubenstein, P. K., Asawaroengchai, C., Nguyen, D. D., Bapna, A., Borsos, Z., de Chaumont Quitry, F., Chen, P., Badawy, D. E., Han, W., Kharitonov, E., Muckenhirn, H., Padfield, D., Qin, J., Rozenberg, D., Sainath, T. N., Schalkwyk, J., Sharifi, M., Ramanovich, M. T., Tagliasacchi, M., Tudor, A., Velimirovic, M., Vincent, D., Yu, J., Wang, Y., Zayats, V., Zeghidour, N., Zhang, Y., Zhang, Z., Zilka, L., and Frank, C. H. Audiopalm: A large language model that can speak and listen. *CoRR*, abs/2306.12925, 2023. doi: 10.48550/ARXIV.2306.12925.

Sadvilkar, N. and Neumann, M. Pysbd: Pragmatic sentence boundary disambiguation. *CoRR*, abs/2010.09657, 2020.

Shazeer, N. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., and and, L. K. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 5998–6008, 2017.

Wahlster, W. *Verbmobil: Foundations of speech-to-speech translation*. Springer, 2000.

Wang, C., Tang, Y., Ma, X., Wu, A., Okhonko, D., and Pino, J. Fairseq S2T: Fast speech-to-text modeling with fairseq. In Wong, D. and Kiela, D. (eds.), *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing: System Demonstrations*, pp. 33–39, Suzhou, China, December 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.aacl-demo.6.

Wang, C., Rivière, M., Lee, A., Wu, A., Talnikar, C., Haziza, D., Williamson, M., Pino, J. M., and Dupoux, E. Voxpopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation. In Zong, C., Xia, F., Li, W., and Navigli, R. (eds.), *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)*, pp. 993–1003. Association for Computational Linguistics, 2021a. doi: 10.18653/V1/2021.ACL-LONG.80.

Wang, C., Wu, A., Gu, J., and Pino, J. Covost 2 and massively multilingual speech translation. In *Interspeech 2021*, pp. 2247–2251, 2021b. doi: 10.21437/Interspeech.2021-2027.

Wang, M., Vu, T., Shareghi, E., and Haffari, G. Conversational simulmt: Efficient simultaneous translation with large language models. *CoRR*, abs/2402.10552, 2024. doi: 10.48550/ARXIV.2402.10552. URL https://doi.org/10.48550/arXiv.2402.10552.

Weiss, R. J., Chorowski, J., Jaitly, N., Wu, Y., and Chen, Z. Sequence-to-sequence models can directly translate foreign speech. In Lacerda, F. (ed.), *18th Annual Conference of the International Speech Communicatio Association, Interspeech 2017*, pp. 2625–2629. ISCA, 2017. doi: 10.21437/INTERSPEECH.2017-503.

Yang, D., Tian, J., Tan, X., Huang, R., Liu, S., Chang, X., Shi, J., Zhao, S., Bian, J., Wu, X., et al. Uniaudio: An audio foundation model toward universal audio generation. *arXiv preprint arXiv:2310.00704*, 2023.

Zeghidour, N., Luebs, A., Omran, A., Skoglund, J., and Tagliasacchi, M. Soundstream: An end-to-end neural audio codec. *IEEE ACM Trans. Audio Speech Lang. Process.*, 30:495–507, 2022. doi: 10.1109/TASLP.2021.3129994.

Zeng, X., Li, L., and Liu, Q. Realtrans: End-to-end simultaneous speech translation with convolutional weighted-shrinking transformer. In Zong, C., Xia, F., Li, W., and Navigli, R. (eds.), *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pp. 2461–2474. Association for Computational Linguistics, 2021. doi: 10.18653/V1/2021.FINDINGS-ACL.218.

Zhang, S., Fang, Q., Guo, S., Ma, Z., Zhang, M., and Feng, Y. Streamspeech: Simultaneous speech-to-speech translation with multi-task learning. In Ku, L., Martins, A., and Srikumar, V. (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pp. 8964–8986. Association for Computational Linguistics, 2024a. doi: 10.18653/V1/2024.ACL-LONG.485.

Zhang, X., Zhang, D., Li, S., Zhou, Y., and Qiu, X. Speechtokenizer: Unified speech tokenizer for speech language models. In *The Twelfth International Conference on Learning Representations*, 2024b.

Zhao, J., Moritz, N., Lakomkin, E., Xie, R., Xiu, Z., Zmolikova, K., Ahmed, Z., Gaur, Y., Le, D., and Fuegen, C. Textless streaming speech-to-speech translation using semantic speech tokens, 2024. URL https://arxiv.org/abs/2410.03298.

A. Appendix

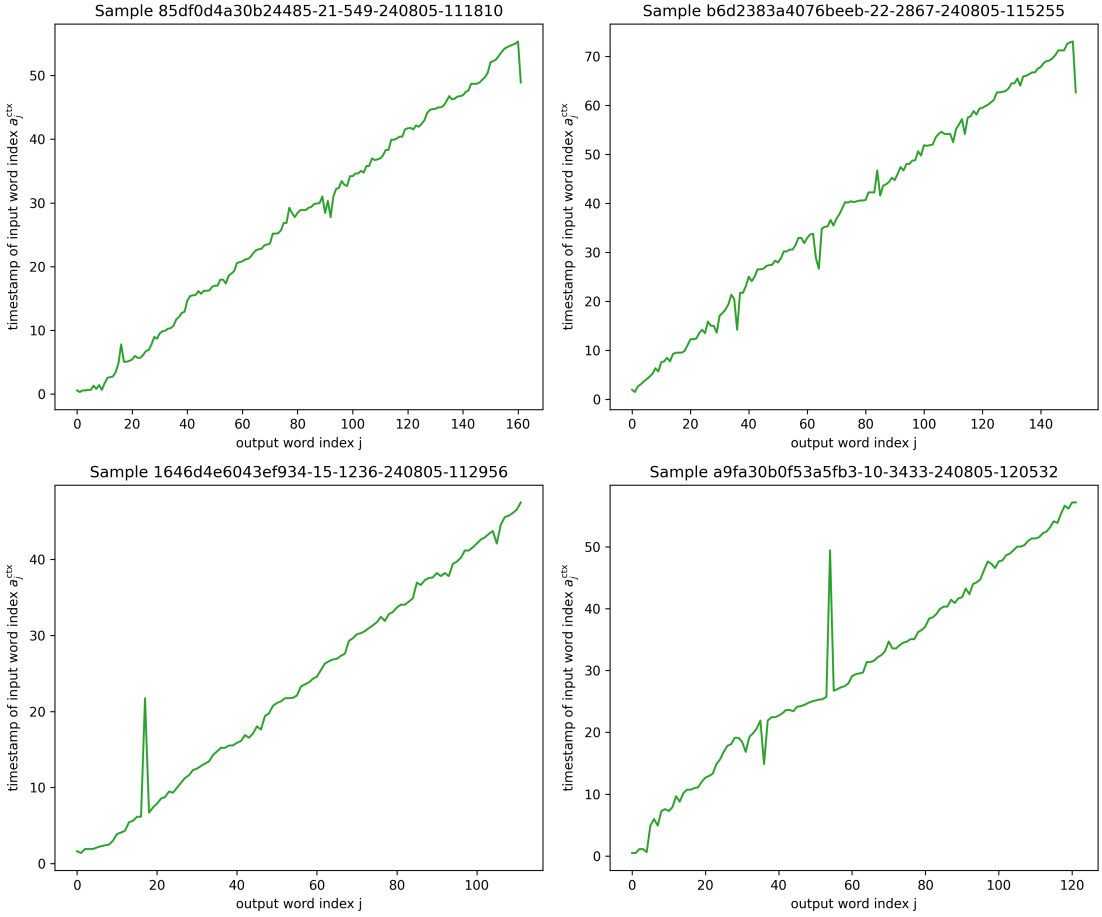

Figure 9. **Example of contextual alignments.** We compute the contextual alignment (a_j^{ctx}) for four different samples and plot the associated input timestamps. Some results as the two at the bottom present extreme spikes interpreted as output words referring to input words very far in the future. These spikes are considered as anomalies and are smoothed out as explained in Section 3.2.2.

Table 6. Architectural hyper-parameters of Hibiki and Hibiki-M.

	Hibiki	Hibiki-M
Temporal Transformer		
Model dimension	2560	2048
Gating dimension	7040	5632
Number of heads	20	16
Number of layers	24	16
Context size	1500	
Depth Transformer		
Model dimension	1024	1024
Gating dimension	2048	2816
Low-rank embeddings dimension	128	-
Number of heads	16	16
Number of layers	4	6
Input / Output space		
Text cardinality	48000	
Audio cardinality per codebook	2048	
Audio codebooks per stream	16	8
Codebooks weight sharing	9 to 16	-
Frame rate	12.5 Hz	
Total number of parameters	2.7 B	1.8 B

Table 7. Objective comparison of Hibiki with Seamless (Barrault et al., 2023) for a English-to-French translation task.

	Short-form (FLEURS En-Fr valid)				Long-form (Audio-NTREX)			
Model	BLEU (↑)	ASR BLEU (↑)	End Offset (↓)	LAAL (↓)	BLEU (↑)	ASR BLEU (↑)	End Offset (↓)	LAAL (↓)
Seamless	39.5	36.6	**2.0**	**2.9**	28.4	23.2	**2.5**	**5.5**
Hibiki	**40.0**	**37.1**	3.9	3.8	**33.9**	**29.6**	4.6	6.7