# Leveraging Large Pre-trained Multilingual Models for High-Quality Speech-to-Text Translation on Industry Scenarios

**Marko Avila**
SYSTRAN by Chapsvision
5 rue Feydeau, 75002 Paris
mavila@chapsvision.com

**Josep Crego**
SYSTRAN by Chapsvision
5 rue Feydeau, 75002 Paris
jcrego@chapsvision.com

## Abstract

Speech-to-Text Translation (S2TT) involves converting spoken language from a source language directly into text in a target language. Traditionally, S2TT systems rely on a sequential pipeline that combines Automatic Speech Recognition (ASR) and Machine Translation (MT) models. However, these systems are prone to error propagation and demand substantial resources to develop and train each component independently. Thus, posing a major challenge in industry settings where cost-effective yet highly accurate S2TT solutions are essential. With the increasing availability of multilingual large pre-trained speech models (LPSM), we propose a parameter-efficient framework that integrates one LPSM with a multilingual MT engine. We evaluate the effectiveness of several well-established LPSMs within this framework, focusing on a real-world industry scenario that involves building a system capable of translating between French, English, and Arabic. The results show that high-quality S2TT systems can be built with minimal computational resources, offering an efficient solution for cross-lingual communication.

## 1 Introduction

Speech-to-Text Translation refers to the process of converting spoken language into written text in a different language, a vital technology for a wide range of applications, including hands-free communication, dictation, video lecture translation, automatic subtitling, and telephone conversations. As globalization expands and the creation of multilingual content increases, the demand for seamless cross-lingual communication becomes more prevalent. S2TT systems address this need effectively by facilitating real-time communication across language barriers.

Traditionally, S2TT systems have been built using a sequential pipeline that combines ASR and MT models (Anastasopoulos et al., 2021; Ney, 1999; Nakamura et al., 2006). In this setup, the ASR component first converts spoken language into text, which is then fed into the MT model for translation. While this method has been effective in taking advantage of improvements in both areas, it has notable drawbacks, such as error propagation, increased training complexity, and longer inference times (Stentiford and Steer, 1988; Waibel et al., 1991). Building and training separate ASR and MT models for each language pair involves substantial computational resources, specialized expertise, and significant time investment, making the development of S2TT systems from scratch a highly resource-intensive endeavor. To address these limitations, the shift in S2TT development is toward end-to-end models, which significantly reduce these issues (Bérard et al., 2016; Bérard et al., 2018; Bentivogli et al., 2021). Nevertheless, even with end-to-end models, significant data and computational resources are still required for their development, leaving resource demands a critical concern.

Recent advancements in deep learning and the increasing availability of large-scale, pre-trained multilingual models for both ASR and MT offer a promising path forward. These models, trained on vast amounts of multilingual data, provide a foundation for developing robust S2TT systems without the need for training from scratch. By leveraging these pre-trained models, it becomes possible to substantially reduce computational and resource demands while maintaining high-quality translations. This approach is especially relevant in industry scenarios where cost-effective yet accurate S2TT solutions are required. Building on this idea, we propose an integrated approach that combines a large pre-trained speech model with a smaller, multilingual NMT system. Unlike larger models, our system is easier to adapt to the specific needs of end users who may not require translations into hundreds of languages. This allows for greater



*Proceedings of the 31st International Conference on Computational Linguistics*, pages 7624–7633
January 19–24, 2025. 

flexibility and customization in multilingual S2TT tasks. This approach greatly reduces computational demands by minimizing the amount of training required, enabling high-quality translations with fewer resources.

The remainder of this paper is organized as follows: Section 2 reviews related work. In Section 3, we present the large pre-trained speech models. We describe the multilingual neural MT network and the hybridization approach implemented. Section 4 gives details of the experimental setup. The results are presented and discussed in Section 5 where we also benchmark our approach against `SeamlessM4T` (Barrault et al., 2023), a state-of-the-art S2TT model. Finally, Section 6 concludes and outlines further research.

## 2 Related Works

Data scarcity and modeling complexity are two major challenges hindering the performance of end-to-end systems (Xu et al., 2023). The first challenge arises from the inherent complexity of speech translation, which combines transcription and translation, making it difficult to optimize a single model for both cross-modal and cross-lingual tasks in a unified step. Secondly, ASR datasets tend to be significantly smaller than MT datasets, and the limited availability of ST datasets further amplifies this discrepancy. To mitigate data scarcity, researchers have adopted techniques like data augmentation (Tsiamas et al., 2023), pretraining (Ao et al., 2021), and knowledge distillation (Liu et al., 2019), which leverage external datasets. In parallel, multi-task learning strategies have been explored to reduce the modeling burden (Zhang et al., 2019; Weiss et al., 2017).

Recent advancements explored multi-tasking in large-scale training, yielding impressive results on Speech-to-Text benchmarks. For example, `Whisper` (Radford et al., 2023) and `SeamlessM4T` (Barrault et al., 2023) incorporate for training a very large amount of multilingual speech data. Building on these large pre-trained speech models, various studies have investigated hybrid systems that leverage such models. In (Khurana et al., 2022), the authors focus on learning multilingual speech-text embeddings at the sentence level, ensuring semantic alignment across languages by aligning embeddings to a multilingual, pre-trained text encoder. A closely related work to ours is presented in (Gow-Smith et al., 2023), where the authors develop a system aimed at improving speech translation quality in low-resource settings coupling two large pre-trained models, an ASR network and an MT network. Similarly, in (Chen et al., 2024), a framework is introduced for leveraging large language models (LLMs) to build S2TT systems, with innovations in model architecture, optimization, ASR-augmented training, multilingual data augmentation, and dual-LoRA optimization.

Our approach differs from these works in that we pair a large pre-trained speech model with a smaller, task-oriented neural MT model. Our main goal being to develop cost-effective, accurate S2TT systems tailored for industry applications.

## 3 Speech-to-Text Translation

This work presents a hybrid solution for parameter-efficient training, integrating speech representation features from a pre-trained speech model into a multilingual Neural Machine Translation (NMT) system. The NMT model, originally designed to generate text in multilingual environments, can be transformed into a multi-modality model capable of performing ASR and multilingual S2TT. The overall hybrid architecture is presented in Figure 1. Our multilingual NMT network (right-most module) receives speech representations (black squares) generated by a speech module (left models). Speech representations are initially reshaped to conform to the word embedding format required by the NMT encoder. Consequently, our S2TT network consists of a speech encoder, a reshape module, and the NMT encoder/decoder network. Note that the speech encoder and reshape module take the place of the word embedding component of the NMT encoder.

This hybrid configuration allows us to convert a multilingual NMT model into a multi-functional system by leveraging data from both ASR and NMT. The hybridization enables the extraction of audio features from various multilingual speech representation models, and the efficiency of parameter training is achieved by only modifying the parameters of the lower layers of the NMT encoder.

### 3.1 Large Pre-trained Speech Models

In our hybrid approach, large pre-trained speech models (LPSM) are kept frozen and used to generate speech representations, which substitute the input word embedding of the NMT network. The

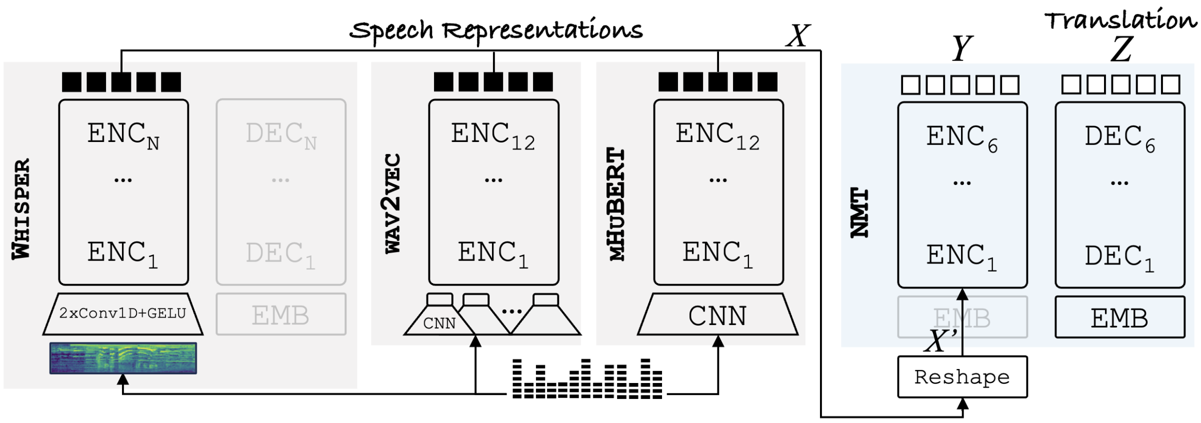


Figure 1: Architecture of our hybrid model combining LPSM and a NMT networks. Three speech networks (left) and one translation network (right). Speech modules produce vector representations, $X$, which are used as input of the NMT network. Representations $X$ are first reshaped to align with the NMT encoder format. Translations are generated from the outputs $Z$ by applying a linear projection followed by a softmax function.

speech representations $X$ consist of the outputs after the $K$ lower encoder layers:

$$\texttt{LPSM}^K_{ENC}(a) = X, \text{ with } X \in \mathcal{R}^{N \times M}$$

with $a$ the audio signal, $N$ the sequence length and $M$ the embedding dimension.

We assess the effectiveness of three distinct LPSMs to generate utterance representations, which we briefly describe in the next lines:

- `wav2vec2`[1] (Baevski et al., 2020) is a speech model that converts raw audio, resampled at 16 kHz, into vector representations for tasks like ASR. Pre-trained on 4.5 million hours of audio using self-supervised learning, it predicts masked segments of the waveform, akin to masked language modeling in NLP. Trained with connectionist temporal classification (CTC), it offers highly efficient and accurate speech recognition with minimal reliance on labeled data. We extract embeddings representations $X$ from the last $k = 12^{th}$ layer, with a variable sequence length $N$, and $M = 768$ corresponding to the hidden layer dimension.

- `mHuBERT-147`[2] (Boito et al., 2024) is a highly efficient multilingual speech representation model trained on $90,430$ hours of open-license speech data across 147 languages. It outperforms larger models, including `wav2vec2`, despite having only $95M$ parameters. This model offers an exceptional balance between high performance and parameter efficiency, making it a promising tool for multilingual speech tasks. In our hybridization work we extracted embeddings $X$ from the last $k = 12^{th}$ layer, with a variable sequence length $N$, and $M = 768$ corresponding to the hidden layer dimension.

- `Whisper`[3] (Radford et al., 2023) is a speech recognition model tailored for multilingual recognition, translation, and language identification. Its Transformer-based architecture integrates multiple speech processing tasks into a single, unified model. It processes audio using an 80-channel log-magnitude Mel spectrogram, resampled at 16 kHz, and employs 30 seconds of context to improve accuracy, implying a fixed sequence length $N = 1500$. The model is released in various sizes. Table 1 provides some details. In our hybridization work we employ the *Medium* version, and use as embedding $X$, the representations resulting from the $K = 6^{th}$ and $K = 24^{th}$ layers of the encoder, with a hidden layer dimension $M = 1024$.

### 3.2 Neural MT Model

Our hybrid approach relies on a multilingual NMT model, which we develop using an in-house imple-

[1] https://huggingface.co/facebook/wav2vec2-base

[2] https://huggingface.co/utter-project/mHuBERT-147

[3] https://huggingface.co/openai/whisper-medium

| *Model* | *Layers* | *Width* | *Heads* | *Size* |
|---|---|---|---|---|
| Tiny | 4 | 384 | 6 | $39M$ |
| Base | 6 | 512 | 8 | $74M$ |
| Small | 12 | 768 | 12 | $244M$ |
| *Medium* | 24 | 1024 | 16 | $769M$ |
| Large | 32 | 1280 | 20 | $1550M$ |

Table 1: Various versions of the `Whisper` model family, detailing the number of layers, embedding width, number of attention heads, and total parameter count for each version.

mentation of the state-of-the-art Transformer architecture[4] (Vaswani et al., 2017). Table 2 gives some details of the network architecture. The model was trained with a mix of open-source bi-texts covering the 4 language pair directions, involving French, English and Arabic. Corpora is obtained from the Opus web site[5]. The training dataset comprises over 110 million sentence pairs, focusing on news, blog, and dialogue data to closely align with the intended use case. The training dataset is balanced as much as possible across all language pair directions to achieve an optimal final checkpoint for each language combination.

| | |
|---|---|
| *size of word embedding* | $1,024$ |
| *size of hidden layers* | $1,024$ |
| *size of inner feed forward layer* | $4,096$ |
| *number of heads* | 16 |
| *number of layers* | 6 |

Table 2: NMT Network specifications.

To enable our model to translate into three languages, we prepend the token $\langle lang \rangle$ at the start of the source stream to indicate the language of the target sentence. During inference, the token guides the model to produce the translation in the specified target language. Source and target training pairs are formatted as follows:

$$source = \langle lang \rangle \; source \; sentence \; \langle eos \rangle$$
$$target = \langle bos \rangle \; target \; sentence \; \langle eos \rangle$$

It is important to note that the NMT model is trained from scratch using written text corpora, which are generally more formal, grammatically correct, and well-structured than speech utterances, typically following standard grammar rules and punctuation. However, the model is ultimately intended to translate speech utterances.

### 3.3 Hybrid S2TT Models

Hybrid models are built upon a standard neural MT network, initially trained for multilingual text translation, coupled with a speech model, as shown in Figure 1. Adaptation is performed to enable our models to perform speech translation and transcription. Note that we fine-tune our models with both speech translations and transcriptions, thus allowing our models to perform both tasks.

As previously discussed, we integrate audio representation features by utilizing the encoder of a speech representation model. The encoder is frozen during the adaptation process. It serves solely for feature extraction and embedding of the speech signal. The LPSMs generate embeddings $X$ with varying embedding lengths, and for fixed (`Whisper`) and variable (`wav2vec2` and `mHuBERT-147`) sequence length. To achieve seamless integration with the NMT encoder, addressing this inconsistency is crucial. We employ the module $Reshape$ to adjust the embeddings output by the speech models into vectors that align with the dimensional requirements of the NMT encoder. The $Reshape$ function is trained in conjunction with the lower $L$ layers of the NMT model's encoder.

#### Reshape Speech Embeddings

To address **embedding dimension mismatch**, a linear projection layer ($M \times 1,024$) is used. Thus, adjusting the size of the embeddings produced by the speech encoder, $M$, to the size of embeddings required by the NMT encoder, $1,024$.

To address the very large **fixed sequence length mismatch** of the `Whisper` encoder, we apply a convolutional layer with a kernel size of 3, a stride of 1 to reduce the sequence length from 1500 to 100 embedding vectors. This allows them to be used as inputs to the NMT encoder.

Models producing variable sequence length embeddings, `wav2vec2` and `mHuBERT-147`, must ensure that do not exceed $1,024$, the maximum sequence length of the NMT model. Larger sequences are filtered out. When working with variable sequence length embeddings, batches containing examples with different sequence length are padded to the batch's maximum sequence length, using a $\langle pad \rangle$ token not considered when computing the loss during training. Figure 2 illustrates

[4] `https://opennmt.net/`
[5] `https://opus.nlpl.eu/`

the Reshape function applied to different speech representations.

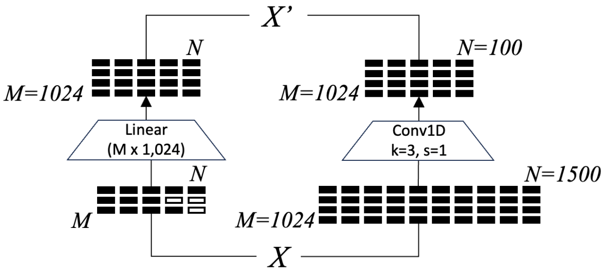


Figure 2: Reshape function applied to speech representations $X$ to ensure embedding size of $1,024$ and shorter sequence lengths. Left path corresponds to wav2vec2 and mHuBERT models; the right path corresponds to the Whisper model (medium size).

To enable our model to translate into three languages, speech reshaped embeddings are appended with the embedding vector of token $\langle lang \rangle$, to specify the target language used. This vector is obtained with the embedding layer of the NMT encoder.

Formally, the following equations describe how the audio signal $a$ is first converted into speech representations $X$ (1). After a reshape operation to adjust its format (2), these are transformed into NMT encoder representations $Y$ (3)[6], which will then be processed by the NMT decoder producing $Z$ (4):

$$X = \texttt{LPSM}^{K}_{ENC}(a) \quad (1)$$
$$X' = \texttt{EMB}(\langle lang \rangle) \cdot \texttt{Reshape}(X) \quad (2)$$
$$Y = \texttt{NMT}_{ENC}(X') \quad (3)$$
$$Z = \texttt{NMT}_{DEC}(Y) \quad (4)$$

where $\cdot$ indicates vector concatenation. Translations are finally generated from $Z$ by applying a linear projection followed by softmax function.

### Tied Speech and Transcription Embeddings

As illustrated in Figure 3 and drawing inspiration from (Khurana et al., 2022), our aim is to generate speech embedding vectors $Y_s$, that are closely aligned with the corresponding transcription embeddings $Y_t$. This approach enables the learning of semantically-aligned multimodal sentence-level representations. By creating speech embeddings that the NMT decoder is already familiar with, we streamline the learning process to produce accurate translations, ultimately improving the system's overall performance. Notice also that the vectors $Y_s$ and $Y_t$ are extracted from the $L$-th layer of the encoder, not necessarily the final layer.

To bias the model towards learning to produce speech embeddings $Y_s$ close to those originally produced for the text transcriptions, $Y_t$, we use an additional term in the loss function that considers the distance between $p = \frac{1}{N_s} \sum_{i=1}^{N_s} Y_{s_i}$ and $q = \frac{1}{N_t} \sum_{i=1}^{N_t} Y_{t_i}$, consisting of average pooling versions of $Y_s$ and $Y_t$ respectively. Thus, we update the loss function with the normalized cosine distance between speech and text sentence representations.

$$\mathcal{L} = \lambda \, \mathcal{L}_{NMT} + (1 - \lambda) \, (1 - cos(p, q))$$

where $\mathcal{L}_{NMT}$ is the regular cross-entropy loss of the NMT network (built for translations) and $\lambda$ is a parameter that indicates the weight of each term in the final loss $\mathcal{L}$. Notice also that training with this extended loss function can only be performed for datasets composed of triplets $\langle$audio speech, transcription, translation$\rangle$.

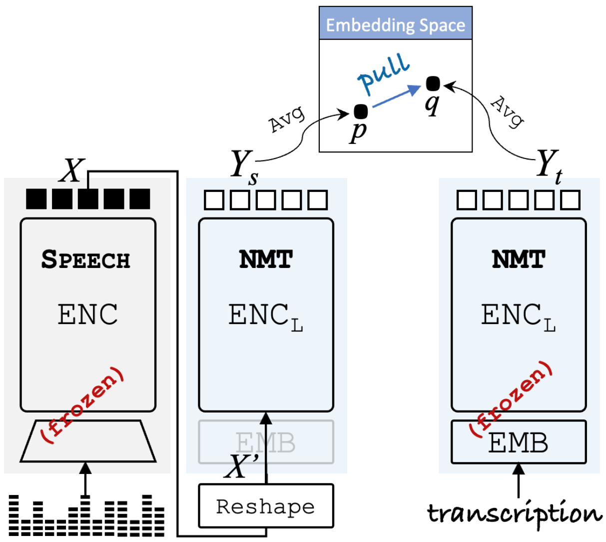


Figure 3: The NMT encoder is fine-tuned to generate $p$, the sentence representation of the audio signal, so that it aligns closely with $q$, the representation of the corresponding transcription. Note that $q$ is produced using a frozen version of the NMT encoder, which was originally trained to work in conjunction with an NMT decoder for producing translations.

It is important to highlight that we utilize two versions of the NMT encoder. The first processes speech embeddings $X$ and generates representations $Y_s$, which are then used by the NMT decoder. The second is a frozen version of the text-based

[6] In training, the $L$ lowest layers are fine-tuned while six are used for inference.

NMT encoder, producing representations $Y_t$.[7] By keeping it frozen and aligning the speech embeddings $Y_s$ with the corresponding transcription embeddings $Y_t$, we facilitate consistency with the representations that the NMT decoder is already familiar with handling.

## 4 Experimental Framework

### 4.1 Datasets

To adapt our models, we use relevant files (including Arabic, French and English) of the open-source dataset **CoVoST 2** (Wang et al., 2021): A large-scale S2TT corpus with $2,900$ hours of speech, covering translations from 21 languages into English, and from English into 15 languages.

Additionally, we use **Fleurs** dataset (Conneau et al., 2022) for testing on the en-fr direction.

Table 3 details the amount of data for each task and language pair. Speech translations are only available in CoVoST 2 for two of our language pairs (fr-en and en-ar), The remaining pairs (fr-ar, en-fr) consist of translations of existing transcriptions.[8] For the fr-ar language pair, we use CoVoST2 fr-en and translate the English transcripts into Arabic. For en-fr, we utilize English audio and translate the corresponding English transcripts into French.

| *Source* | *Lang* | *Train* | *Test* |
|---|---|---|---|
| ASR | | | |
| CoVoST2 | fr | 200,000 | 15,531 |
| CoVoST2 | en | 200,000 | 14,760 |
| S2TT | | | |
| CoVoST2 | fr-en | 200,000 | 14,760 |
| CoVoST2* | fr-ar | 200,000 | - |
| CoVoST2* | en-fr | 200,000 | - |
| CoVoST2 | en-ar | 200,000 | 15,531 |
| Fleurs | en-fr | - | 3,643 |

Table 3: Corpus Statistics. Datasets used for each task, including source, language and the number of training, and test sentences (or utterances). Machine-translated datasets are marked with an asterisk (*).

In summary, we use 2,239 hours of speech for training and 182 hours for testing. It is important to note that the ASR and S2TT training datasets are imbalanced, with the S2TT dataset containing roughly twice as many examples as the ASR dataset. Additionally, while we built an S2TT model for 4 language pair directions, we only evaluated it on 3, as no test set was available for the fr-ar direction.

### 4.2 Networks

The **NMT** model training work employs a single NVIDIA V100 GPU (32GB). We use the lazy Adam algorithm (Kingma and Ba, 2015) for optimization. We set warm-up steps to $4,000$ and update learning rate for every 8 iterations. We limit the source and target sentence lengths to 150 tokens based on BPE (Sennrich et al., 2016) preprocessing. A total of $28K$ BPE merge operations are separately computed for each language. We finally use a joint Arabic, French and English vocabulary of $50K$ tokens. In inference we use a beam size of 5.

Our **Hybrid** models are trained using a single NVIDIA V100 GPU (32GB) during up to $500,000$ updates, with a maximum batch size of 400 source tokens and updates of the model after accumulating 25 batches. We validate every $5,000$ updates and perform early stopping on a separate validation set excluded from the training set.

## 5 Results

Table 4 presents a summary of results for several networks and configurations. BLEU (Post, 2018) and WER[9] are used as metrics for S2TT and ASR evaluation, respectively. WER scores are computed over *normalized* transcriptions[10]. Bold face is used to outline best scores of each test set.

The columns *LPSM Enc* and *Dec* show the number of layers used during inference by the Speech Encoder and Decoder, respectively. Columns *NMT Enc* and *Dec* indicate the number of fine-tuned encoder/decoder layers in the NMT model. For inference, 6 encoder and 6 decoder layers of the NMT model are consistently utilized. Column *Size* indicates the number of model parameters used by each system during inference. The *Avg* columns (with gray background) display the average results of the reference S2TT and ASR tests. Column *Avg1* indicates the average translation BLEU scores for CoVoST2 in-domain test sets (en-ar and fr-en)

[7]This second version of the NMT encoder is only used for training, not employed at inference time.

[8]Machine translations are performed using the open-source NLLB 3.3B model `https://huggingface.co/facebook/nllb-200-3.3B`

[9]`https://huggingface.co/spaces/evaluate-metric/wer`

[10]Normalization performed with `BasicTextNormalizer` of the `transformers.models.whisper` library.

| Model | LPSM Inf | | NMT Opt | | Size | BLEU↑ | | | | | WER↓ | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Enc | Dec | Enc | Dec | | en-ar | fr-en | Avg1 | en-fr | Avg2 | en | fr | Avg |
| *Cascade* | | | | | | | | | | | | | |
| **whisper+nllb** | 24 | 24 | - | - | 4.1B | 19.40 | 33.46 | 26.43 | 43.91 | 32.26 | 10.34 | 14.96 | 12.65 |
| **whisper+nmt** | 24 | 24 | - | - | 997M | 19.72 | 31.37 | 25.55 | 41.22 | 30.77 | 10.34 | 14.96 | 12.65 |
| Whisper *fine-tunned* | | | | | | | | | | | | | |
| **whisper** | 24 | 24 | - | - | 769M | 16.10 | 33.83 | 24.97 | 31.19 | 27.02 | 17.21 | 14.15 | 15.68 |
| *SOTA* | | | | | | | | | | | | | |
| **seamless_m** | - | | | | 1.2B | 21.61 | 39.12 | 30.37 | 37.47 | 32.73 | 8.15 | 12.20 | 10.18 |
| **seamless_l** | | | | | 2.3B | 24.30 | 40.72 | 32.51 | 42.77 | 35.93 | 6.79 | 11.14 | 8.97 |
| *Hybrid (this work)* | | | | | | | | | | | | | |
| **wav2vec-nmt** | 12 | - | 2 | - | 271M | 15.00 | 21.38 | 18.19 | 26.14 | 20.84 | 30.12 | 37.27 | 33.70 |
| | 12 | - | 4 | - | 271M | 15.39 | 24.32 | 19.86 | 25.77 | 21.83 | 27.94 | 30.36 | 29.15 |
| | 12 | - | 6 | - | 271M | 15.41 | 24.34 | 19.88 | 24.90 | 21.55 | 27.10 | 28.72 | 27.91 |
| **mhubert-nmt** | 12 | - | 2 | - | 271M | 16.62 | 31.41 | 24.02 | 24.88 | 24.30 | 22.20 | 18.06 | 20.13 |
| | 12 | - | 4 | - | 271M | 17.44 | 32.47 | 24.96 | 25.24 | 25.10 | 20.51 | 15.69 | 18.10 |
| | 12 | - | 6 | - | 271M | 16.75 | 31.78 | 24.27 | 24.29 | 24.27 | 20.16 | 15.43 | 17.80 |
| **whisper-nmt** | 6 | - | 2 | - | 263M | 10.74 | 26.34 | 18.54 | 18.92 | 18.67 | 34.58 | 25.65 | 30.12 |
| | 24 | - | 2 | - | 488M | 21.48 | 35.73 | 28.61 | 30.27 | 29.20 | 14.22 | 12.72 | 13.47 |
| | 24 | - | 4 | - | 488M | 21.74 | 35.92 | 28.83 | 30.40 | 29.35 | 13.95 | 12.33 | 13.14 |
| | 24 | - | 6 | - | 488M | 21.80 | 35.90 | 28.85 | 30.30 | 29.33 | 13.51 | 12.06 | 12.79 |
| | 24 | - | 6 | 6 | 488M | 22.41 | 35.77 | 29.10 | 30.29 | 29.50 | 13.54 | 11.31 | 12.43 |
| **whisper-nmt**$^{tied}$ | 24 | - | 2 | - | 488M | 21.55 | 35.57 | 28.56 | 29.39 | 28.83 | 14.46 | 12.76 | 13.61 |

Table 4: Translation (BLEU) and recognition (WER) results across various model configurations. The column *LPSM Inf* specifies the number of encoder/decoder layers during inference, while *NMT Opt* shows the number of NMT encoder/decoder layers optimized during training. The *Size* column denotes the total number of parameters used during inference.

while column *Avg2* averages all translation test set results.

System **whisper+nmt** is a cascade system performing transcriptions with the LPSM followed by the NMT network.

System **whisper** involves fine-tuning the entire Whisper model for both ASR and S2TT tasks using exactly the same training datasets than are used for the rest of optimizations. Notably, this is the only configuration where the LPSM model is fine-tuned, leading to significantly longer training times (nearly two weeks) and with BLEU results behind those of the cascade model.

Systems **seamless_m** and **seamless_l** are respectively the medium and large versions of the same network (SeamlessM4T). As anticipated, they achieve state-of-the-art results in both tasks (averaging $32.73$ and $35.93$ respectively). However, they are the models with the largest number of parameters, requiring the most resources.

The next set of results correspond to our hybrid systems **whisper-nmt**, **mhubert-nmt** and **wav2vec-nmt**, which couple the evaluated LPSMs with our NMT model. Different configurations are evaluated for each. Hybrid models are notably smaller in size, and with the LPSMs kept frozen, they require minimal training iterations. Fine-tuning the hybrid models with our training dataset took between 1 and 5 days, depending on the number of NMT parameters optimized.

Regarding **whisper-nmt** and following (Pasad et al., 2021; Gow-Smith et al., 2023) which argue that some speech representation models tend to have a higher abstraction from the speech signal in the middle layers, we evaluate using the $6^{th}$ encoding layer of the Whisper model as feature extractor. However, the best results are achieved when **whisper-nmt** employs the full encoder to produce speech representations $Y$ with all its 24 layers. Varying the number of fine-tuned NMT encoder layers (2, 4, or 6) results in a modest impact, with differences of less than 1 BLEU point across all hybrid networks. The **mhubert-nmt** and **wav2vec-nmt** systems consistently produce significantly lower BLEU scores compared to the **whisper-nmt** system.

Optimizing the NMT decoder fully has little impact on the average BLEU of $0.17$ points. Concerning **whisper-nmt**$^{tied}$, which employs an alternative loss function to align the NMT encoder's speech representations with those generated by the same encoder for corresponding transcriptions, the results do not improve over the system without tied representations.[11]

The best hybrid results are around 3 BLEU points lower than **seamless_m** and comparable to those of the cascade system. It's important to note that the hybrid system is significantly smaller, with over four times fewer parameters than the **seamless_l** model and half the size of both the cascade and **seamless_m** models. Additionally, it was trained with substantially fewer resources than the seamless models.

Note that for the en-ar and fr-en translation directions, our best hybrid system's results are closer to the top scores, trailing by around 3.5 BLEU points. In contrast, for en-fr, the hybrid system lags more than 10 BLEU points behind. This discrepancy arises because we fine-tune our hybrid models using the CoVoST 2 dataset, which is also used for en-ar and fr-en testing, while en-fr testing data is comes from the Fleurs dataset. Our smaller hybrid systems are more adversely affected by domain shifts compared to the larger models.

Regarding the ASR evaluation, **seamless_l** obtains best results (8.97) with less than 3 WER points than those obtained by the original `Whisper` **whisper+nmt** (12.65). When `Whisper` is optimized to achieve translation abilities its ASR performance is lowered with a WER score of 15.68.

With respect to hybrid models, similar to the translation accuracy results, both **wav2vec-nmt** and **mhubert-nmt** show poorer performance compared to **whisper-nmt**, which achieves its best results with the optimization of 6 encoder and 6 decoder layers, reaching an average WER score of 12.43. Notably, the WER for French speech is particularly impressive (11.31), comparable to the results obtained by the best system **seamless_l** (11.14) and more than 3 points lower than the WER achieved by the original `Whisper` model (14.96).

Finally, Table 5 compares some of the systems presented in this work in terms of model size (number of parameters) and inference time, with results reported relative to our **whisper-nmt** network. Note that for inference, we use Hugging Face[12] libraries on a single NVIDIA V100 GPU (32GB) with comparable inference settings. As shown, the system presented in this work achieves the best efficiency, primarily due to its use of the smallest number of parameters.

| *Model* | *Size* | *Time* |
|---|---|---|
| **whisper-nmt** | ×1.0 | ×1.0 |
| **whisper+nllb** | ×8.4 | ×4.0 |
| **whisper** | ×1.6 | ×1.1 |
| **seamless_m** | ×2.5 | ×2.2 |
| **seamless_l** | ×4.7 | ×4.3 |

Table 5: Number of parameters (*Size*) and inference time (*Time*) of different networks reported relative to the **whisper-nmt** network results.

## 6 Conclusions and Further Work

We developed a Speech-to-Text Translation system that minimizes the need for extensive computational resources and large datasets. By leveraging pre-trained models and implementing efficient hybrid approaches, we evaluated several LPSMs in a real-world industry scenario, demonstrating that highly accurate S2TT systems can be built with minimal resources, making them more accessible without the need for extensive infrastructure. Furthermore, our system has also been shown to deliver accurate ASR performance.

We are currently addressing the domain shift issue observed in our NMT model. Our plan is to develop a more robust model using a broader range of bilingual texts, in contrast to the current approach, which relied on corpora closely matching the speech style. We plan to develop a fast inference library to implement the proposed hybridization, ensuring efficient execution of our system on both CPU and GPU platforms, a crucial feature for industrial applications. We are also exploring a system capable of both transcription and translation by means of a synchronized dual decoder.

## Acknowledgments

This work has been funded by the French Ministry of Defense through the DGA-RAPID 2022190955, COMMUTE project.

## References

Antonios Anastasopoulos, Ondřej Bojar, Jacob Bremerman, Roldano Cattoni, Maha Elbayad, Marcello Federico, Xutai Ma, Satoshi Nakamura, Matteo Negri, Jan Niehues, Juan Pino, Elizabeth Salesky, Sebastian Stüker, Katsuhito Sudoh, Marco Turchi, Alexander Waibel, Changhan Wang, and Matthew Wies-

[11] The results for the tied embeddings experiment were obtained after fewer learning iterations due to time constraints. We will present results with a comparable number of iterations in the camera-ready version of the paper.

[12] https://huggingface.co/

ner. 2021. Findings of the IWSLT 2021 Evaluation Campaign. In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 1–29, Bangkok, Thailand (online). Association for Computational Linguistics.

Junyi Ao, Rui Wang, Long Zhou, Shujie Liu, Shuo Ren, Yu Wu, Tom Ko, Qing Li, Yu Zhang, Zhihua Wei, Yao Qian, Jinyu Li, and Furu Wei. 2021. Speecht5: Unified-modal encoder-decoder pre-training for spoken language processing. In *10.48550/arXiv.2110.07205*.

Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: a framework for self-supervised learning of speech representations. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Loïc Barrault, Yu-An Chung, Mariano Meglioli, David Dale, Ning Dong, Paul-Ambroise Duquenne, Hady Elsahar, Hongyu Gong, Kevin Heffernan, John Hoffman, Christopher Klaiber, Pengwei Li, Daniel Licht, Jean Maillard, Alice Rakotoarison, Kaushik Sadagopan, Guillaume Wenzek, and Skyler Wang. 2023. Seamlessm4t-massively multilingual & multimodal machine translation. *10.48550/arXiv.2308.11596*.

Luisa Bentivogli, Mauro Cettolo, Marco Gaido, Alina Karakanta, Alberto Martinelli, Matteo Negri, and Marco Turchi. 2021. Cascade versus direct speech translation: Do the differences still make a difference? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2873–2887, Online. Association for Computational Linguistics.

Alexandre Bérard, Olivier Pietquin, Laurent Besacier, and Christophe Servan. 2016. Listen and Translate: A Proof of Concept for End-to-End Speech-to-Text Translation. In *NIPS Workshop on end-to-end learning for speech and audio processing*, Barcelona, Spain.

Marcely Zanon Boito, Vivek Iyer, Nikolaos Lagos, Laurent Besacier, and Ioan Calapodescu. 2024. mHuBERT-147: A Compact Multilingual HuBERT Model. In *Interspeech 2024*.

Alexandre Bérard, Laurent Besacier, Ali Can Kocabiyikoglu, and Olivier Pietquin. 2018. End-to-end automatic speech translation of audiobooks. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6224–6228.

Xi Chen, Songyang Zhang, Qibing Bai, Kai Chen, and Satoshi Nakamura. 2024. LLaST: Improved end-to-end speech translation system leveraged by large language models. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 6976–6987, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna. 2022. Fleurs: Few-shot learning evaluation of universal representations of speech. *arXiv preprint arXiv:2205.12446*.

Edward Gow-Smith, Alexandre Berard, Marcely Zanon Boito, and Ioan Calapodescu. 2023. NAVER LABS Europe's multilingual speech translation systems for the IWSLT 2023 low-resource track. In *Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023)*, pages 144–158, Toronto, Canada (in-person and online). Association for Computational Linguistics.

Sameer Khurana, Antoine Laurent, and James Glass. 2022. Samu-xlsr: Semantically-aligned multimodal utterance-level cross-lingual speech representation. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1493–1504.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Y. Liu, H. Xiong, J. Zhang, Z. He, H. Wu, H. Wang, and C. Zong. 2019. End-to-end speech-translation with knowledge distillation. *Proc. Interspeech 2019*, pages 1128–1132.

Satoshi Nakamura, Konstantin Markov, Hiromi Nakaiwa, Gen ichiro Kikui, Hisashi Kawai, Takatoshi Jitsuhiro, Jin-Song Zhang, Hirofumi Yamamoto, Eiichiro Sumita, and Seiichi Yamamoto. 2006. The atr multilingual speech-to-speech translation system. *IEEE Transactions on Audio, Speech, and Language Processing*, 14:365–376.

Hermann Ney. 1999. Speech translation: Coupling of recognition and translation. In *10.1109/ICASSP.1999.758176*, volume 1, pages 517–520 vol.1.

Ankita Pasad, Ju-Chieh Chou, and Karen Livescu. 2021. Layer-wise analysis of a self-supervised speech representation model. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 914–921.

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

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual*

*Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Fred Stentiford and M.G. Steer. 1988. Machine translation of speech. *British Telecom Technology Journal*, 6:116–123.

Ioannis Tsiamas, José Fonollosa, and Marta Costa-jussà. 2023. SegAugment: Maximizing the utility of speech translation data with segmentation-based augmentations. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8569–8588, Singapore. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

A. Waibel, A.N. Jain, A.E. McNair, H. Saito, A.G. Hauptmann, and J. Tebelskis. 1991. Janus: a speech-to-speech translation system using connectionist and symbolic processing strategies. In *[Proceedings] ICASSP 91: 1991 International Conference on Acoustics, Speech, and Signal Processing*, pages 793–796 vol.2.

Changhan Wang, Anne Wu, Jiatao Gu, and Juan Pino. 2021. Covost 2 and massively multilingual speech translation. In *Interspeech 2021*, pages 2247–2251.

Ron J. Weiss, Jan Chorowski, Navdeep Jaitly, Yonghui Wu, and Z. Chen. 2017. Sequence-to-sequence models can directly translate foreign speech. In *Interspeech*.

Chen Xu, Rong Ye, Qianqian Dong, Chengqi Zhao, Tom Ko, Mingxuan Wang, Tong Xiao, and Jingbo Zhu. 2023. Recent advances in direct speech-to-text translation. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, pages 6796–6804. International Joint Conferences on Artificial Intelligence Organization. Survey Track.

Pei Zhang, Niyu Ge, Boxing Chen, and Kai Fan. 2019. Lattice transformer for speech translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6475–6484, Florence, Italy. Association for Computational Linguistics.