Speech Vecalign: an Embedding-based Method for Aligning Parallel Speech Documents

Chutong Meng[*]
George Mason University
cmeng2@gmu.edu

Philipp Koehn
Johns Hopkins University
phi@jhu.edu

Abstract

We present Speech Vecalign, a parallel speech document alignment method that monotonically aligns speech segment embeddings and does not depend on text transcriptions. Compared to the baseline method Global Mining (Duquenne et al., 2023a), a variant of speech mining, Speech Vecalign produces longer speech-to-speech alignments. It also demonstrates greater robustness than Local Mining, another speech mining variant, as it produces less noise. We applied Speech Vecalign to 3,000 hours of unlabeled parallel English-German (En-De) speech documents from VoxPopuli, yielding about 1,000 hours of high-quality alignments. We then trained En-De speech-to-speech translation models on the aligned data. Speech Vecalign improves the En-to-De and De-to-En performance over Global Mining by 0.37 and 0.18 ASR-BLEU, respectively. Moreover, our models match or outperform SpeechMatrix model performance, despite using 8 times fewer raw speech documents.[1]

1 Introduction

Speech-to-speech translation (S2ST) is the task of translating speech in one language into speech in another language. Conventional S2ST systems concatenate automatic speech recognition (ASR), machine translation (MT), and text-to-speech (TTS) models (Lavie et al., 1997; Nakamura et al., 2006; Wahlster, 2013). These components can be trained individually with datasets for the different components. Direct S2ST models, which translate source speech into target spectrograms or discrete units with a single architecture, have been recently proposed to alleviate error propagation and to reduce inference latency (Jia et al., 2019; Lee et al., 2022a).

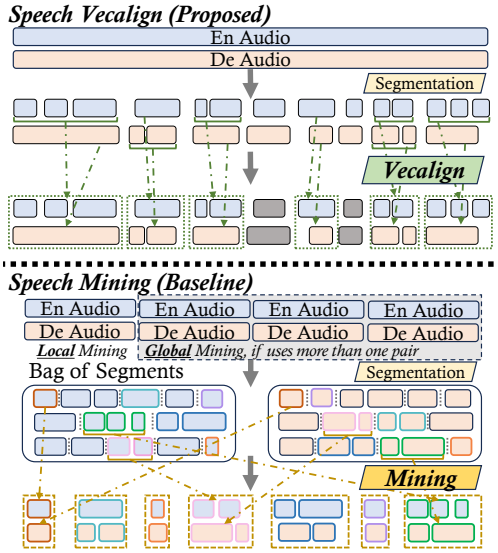


Figure 1: A comparison between Speech Vecalign (above) and speech mining (below). Speech Vecalign aligns each pair of speech documents individually and aligns segments in chronological order, while speech mining aligns bags of segments and ignores the structure of parallel speech documents.

Despite the advantages, performance of direct models is limited by the amount of speech-to-speech aligned data, which is much more scarce than the data used for components of cascaded systems.

There have been efforts to automatically curate alignments from multilingual *speech document*s. In this paper, we define a *speech document* as a file containing more than one utterance and typically comprising several paragraphs, analogous to a *text document*. VoxPopuli (Wang et al., 2021a) is one such corpus containing a large number of *parallel* speech documents, which are pairs of documents that have the same content but differ in language.

Speech-to-speech alignment methods align short speech clips called *segment*s, and can be either transcription-based or transcription-free. When transcriptions are available, segments in parallel speech documents can be aligned through speech-to-text and text-to-text alignments. Inspired by

[*]Work done at Johns Hopkins University.

[1]Data and code are available at https://github.com/mct10/Speech-Vecalign.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 16478–16494

text mining (Schwenk et al., 2021), speech mining (Duquenne et al., 2021, 2023a) was proposed as a transcription-free method that aligns speech segments by finding segment pairs with the highest embedding similarity. It scales well as it does not rely on the availability of text transcriptions. When speech mining is applied to a large amount of speech documents, as in all previous work, it is referred to as **Global Mining**. Another variant, **Local Mining**, which applies speech mining to a single pair of parallel speech documents, has not been well explored. As we formally define in Section 2, both Global Mining and Local Mining treat documents as bags of unordered segments.

Since speech mining methods do not leverage the document pair structure, we wonder, **can we obtain better alignments by aligning speech segments within document pairs and preserving their time order?** This allows us to utilize the extra knowledge that (1) segments within parallel document pairs are likely to be translations of each other, and (2) segment pairs right next to already aligned pairs are also likely to be aligned. We draw inspiration from parallel *text* document alignment methods, which have been popular to create sentence-aligned bitext for training MT systems. Unlike mining, they align sentences for each document pair while maintaining the sentence order. Our work is based on the text alignment method Vecalign (Thompson and Koehn, 2019), which aligns parallel sentences by applying fast dynamic time warping (Salvador and Chan, 2007) to sentence embeddings. With the advances of extending sentence embeddings to the speech modality (Duquenne et al., 2021), we can readily apply Vecalign to parallel speech documents.

In this paper, we introduce Speech Vecalign, a method that aligns parallel speech documents using speech segment embeddings. Instead of mining from bags of segments, our method aligns individual document pairs and maintains the chronological order of segments, as illustrated in Figure 1. Additional preprocessing and postprocessing strategies are applied to improve alignment quality. We compare Speech Vecalign with Local Mining and Global Mining and show that Speech Vecalign produces higher-quality alignments. We further provide extensive analysis for all three methods, which could be useful for future research.

2 Speech Mining Overview

We formally describe the speech mining methods in this section. Other related work is in Appendix A.

Speech Mining, first proposed by Duquenne et al. (2021), encodes speech segments into language- and modality-agnostic fixed-size embeddings, and then uses margin-based similarity search (Artetxe and Schwenk, 2019a) to find the closest embedding pairs. Depending on the search scope, it can be categorized as Global Mining or Local Mining.

Raw data. The input data is a list of speech documents $X = [X_1, X_2, \ldots, X_n]$ in the source language and a list $Y = [Y_1, Y_2, \ldots, Y_m]$ in the target language, where n and m are the numbers of documents. Each document can contain between a few seconds to a few hours of speech.

Speech segmentation. Voice activity detection (VAD) is applied to each document to obtain short segments, typically lasting a few seconds. For instance, X_i is segmented into $X_i = [x_1^i, x_2^i, \ldots, x_{n_i}^i]$, where n_i is the number of segments. To have segments at different granularities, consecutive segments are progressively concatenated. X_i becomes $\tilde{X}_i = \left[\tilde{x}_1^i, \tilde{x}_2^i, \ldots, \tilde{x}_{\tilde{n}_i}^i\right]$, where $\tilde{x}$ denotes a concatenated segment and $\tilde{n}_i$ denotes the number of resulting segments. The same process applies to Y_j, producing $\tilde{Y}_j$.

Speech segment embedding. Each segment is encoded into a fixed-size embedding using an embedding model. The segment embeddings for $\tilde{X}_i$ are represented as $E_{\tilde{X}_i} = \left[e_1^{\tilde{X}_i}, e_2^{\tilde{X}_i}, \ldots, e_{\tilde{n}_i}^{\tilde{X}_i}\right]$. Similarly, the segments in $\tilde{Y}_j$ are encoded as $E_{\tilde{Y}_j}$.

Bag of embeddings. In *Global Mining*, embeddings are grouped by **language**. We define $G_X = \left\{E_{\tilde{X}_1}, E_{\tilde{X}_2}, \ldots, E_{\tilde{X}_n}\right\}$ and $G_Y = \left\{E_{\tilde{Y}_1}, E_{\tilde{Y}_2}, \ldots, E_{\tilde{Y}_m}\right\}$, where G_X collects all segment embeddings in the source language and G_Y collects those in the target language. In *Local Mining*, embeddings are grouped by **document pairs**. Suppose there are s parallel documents, with X_i paired with Y_i for $1 \leq i \leq s$. Documents without a parallel one are ignored. In this case, $E_{\tilde{X}_i}$ and $E_{\tilde{Y}_j}$ are bags of embeddings themselves.

Embedding alignment. Speech mining is performed by finding the most similar embedding pairs between two bags of segment embeddings. The margin-based similarity, or margin-score, between

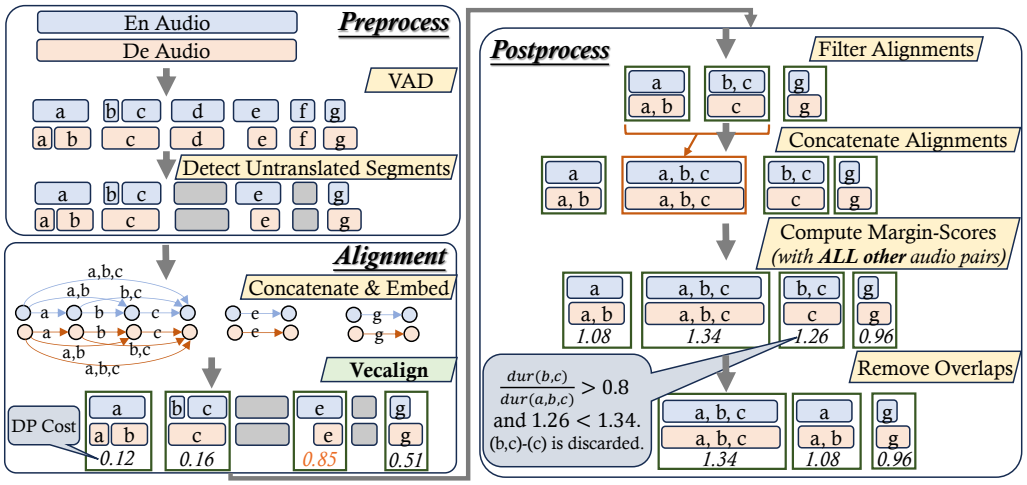


Figure 2: An illustration of the complete Speech Vecalign pipeline using a simple example. Each pair of speech documents need to go through 3 steps: Speech Preprocessing (Section 3.1), Segment Alignment (Section 3.2), and Alignment Postprocessing (Section 3.3).

any two embeddings a and b is computed as

$$\text{sim}(a,b) = \frac{\cos(a,b)}{\sum_{z\in \text{NN}_k(a)} \frac{\cos(a,z)}{2k} + \sum_{z\in \text{NN}_k(b)} \frac{\cos(b,z)}{2k}} \quad (1)$$

where a and b are in different languages and $\text{NN}_k(a)$ denotes k nearest neighbors of a in the other language. The denominator combats the hubness problem. A higher margin-score indicates better quality. Then, the mining function for embedding a from a bag of embeddings B is

$$\text{mine}(a,B) = \underset{b\in B}{\text{argmax}}\ \text{sim}(a,b) \quad (2)$$

More generally, given two bags of embeddings $U = \{u_1, u_2, \ldots, u_{l_u}\}$ and $V = \{v_1, v_2, \ldots, v_{l_v}\}$, where l_u and l_v are number of embeddings, the collection of all speech mining alignments is

$$\begin{aligned}&\text{align}(U,V)\\ &= \{(u_1, \text{mine}(u_1,V)), \ldots, (u_{l_u}, \text{mine}(u_{l_u},V))\}\\ &\bigcup \{(\text{mine}(v_1,U), v_1), \ldots, (\text{mine}(v_{l_v},U), v_{l_v})\}\end{aligned} \quad (3)$$

Finally, we define Local Mining and Global Mining as

$$\text{Global-Mine}(X,Y) = \text{align}(G_X, G_Y) \quad (4)$$

$$\text{Local-Mine}(X,Y) = \bigcup_{i=1}^{s} \text{align}(E_{\tilde{X}_i}, E_{\tilde{Y}_i}) \quad (5)$$

3 Proposed Method: Speech Vecalign

The Speech Vecalign pipeline consists of three steps: speech preprocessing (Section 3.1), segment alignment with Vecalign (Section 3.2), and alignment postprocessing (Section 3.3). An illustration of our method is shown in Figure 2.

3.1 Speech Preprocessing

Speech preprocessing consists of document segmentation and detection of identical untranslated segments.

Segmentation. Same as speech mining, we first segment each speech document by VAD. We apply Silero VAD (Silero Team, 2021).

Detection of identical untranslated segments. As mentioned by Wang et al. (2021a), some source and target segments contain identical untranslated content due to recording issues. We introduce this additional step to detect such pairs of segments *prior to* applying the alignment algorithms, in order to make sure they are not aligned.

To find potentially identical untranslated segment pairs, we use a *location heuristic* that they tend to locate in roughly the same position within the source and target documents. For instance, within each pair of parallel documents, for a source segment x_a^i spanning timestamp $s_{x_a}^i$ to $e_{x_a}^i$, we search for a target segment y_b^i whose midpoint $\frac{s_{y_b}^i + e_{y_b}^i}{2}$ is closest to $\frac{s_{x_a}^i + e_{x_a}^i}{2}$, midpoint of x_a^i, since the untranslated target segment is very likely to have a similar time span ($s_{y_b}^i \approx s_{x_a}^i$, $e_{y_b}^i \approx e_{x_a}^i$).

If the two segments have both similar durations and filterbank features, we classify them as identical. For durations, we compute the time difference.

For filterbank feature, we compute Equation 6:

$$\text{sim}(\mathbf{A}, \mathbf{B}) = \min_{i \in \{1,\ldots,T_2-T_1+1\}} \left\{ \frac{1}{NT} \|\mathbf{A} - \mathbf{B}_{:,\mathbf{i}:\mathbf{i}+\mathbf{T_1}-\mathbf{1}}\|_F^2 \right\} \quad (6)$$

$\mathbf{A} \in \mathbb{R}^{N \times T_1}$ and $\mathbf{B} \in \mathbb{R}^{N \times T_2}$ are filterbank features[2] with $N = 80$ mel-frequency bins. T_1 and T_2 are numbers of frames. Without loss of generality, we assume $T_1 \leq T_2$. $\mathbf{B}_{:,\mathbf{i}:\mathbf{i}+\mathbf{T_1}-\mathbf{1}} \in \mathbb{R}^{N \times T_1}$ denotes a slice of $\mathbf{B}$ from frame i to frame $i + T_1 - 1$. $\|\cdot\|_F$ denotes the Frobenius norm, and $\frac{1}{NT}\|\mathbf{A} - \mathbf{B}_{:,\mathbf{i}:\mathbf{i}+\mathbf{T_1}-\mathbf{1}}\|_F^2$ is the mean squared error between $\mathbf{A}$ and a slice of $\mathbf{B}$. We check if $\mathbf{A}$ could be identical to any slice of $\mathbf{B}$, tolerating the leading and/or trailing noise or silence frames in $\mathbf{B}$.

We empirically determine 0.1 second and $\text{sim}(\mathbf{A}, \mathbf{B}) = 5.0$ as the thresholds for duration and filterbank similarities. Note that we cannot depend on speech embeddings for detection, as speech encoders are multilingual and their embeddings are language-agnostic.

3.2 Speech Segment Alignment

We perform segment alignment based on the similarity between speech segment embeddings. Unlike speech mining, which solely relies on similarity scores, we use a dynamic programming (DP) algorithm to align segments in chronological order.

Segment concatenation. Speech segments do not necessarily correspond to complete sentences. Same as speech mining, we first progressively concatenate each segment with the subsequent ones. Each concatenated segment can contain up to 5 original segments and span a maximum of 20 seconds.

Obtaining segment embeddings. After concatenations, we obtain speech segment embeddings using SpeechLASER models (Duquenne et al., 2021, 2023a). Identical untranslated segments detected in Section 3.1, along with all concatenated segments that include them, are skipped and replaced with 0-valued vectors.

Applying Vecalign to embeddings. We follow Thompson and Koehn (2019) to define the cost of aligning two segment embeddings, which serves as the cost function for DP:

$$c(x, y) = \frac{(1 - \cos(x, y))\text{nSegs}(x)\text{nSegs}(y)}{\sum_{s=1}^{S} \frac{1-\cos(x, y_s)}{2S} + \sum_{s=1}^{S} \frac{1-\cos(x_s, y)}{2S}} \quad (7)$$

[2]We use torchaudio.compliance.kaldi.fbank.

x and y are segment embeddings. $\cos(\cdot, \cdot)$ computes the cosine similarity. x_s and y_s are uniformly sampled source and target embeddings, and S is the sample size. $\text{nSegs}(x)$ is used to denote the number of original segments in x, which penalizes aligning long concatenations.

The embedding alignment algorithm is recursive DP. Given a document pair and corresponding embeddings, the algorithm recursively averages every two consecutive embeddings, halving the sequence length until it reaches a small threshold. At the bottom level, standard DP is applied to obtain an initial alignment. Subsequently, at each recursion level bottom-up, DP refines the alignment by searching within a small window around the alignment path from the previous level. By constraining the search space and reducing the sequence length at each level, the algorithm achieves a linear time and space complexity. The recursive DP algorithm runs on CPU and takes a few seconds on average per document pair. We direct the readers to Thompson and Koehn (2019) for a complete description.

Because of DP, the resultant alignments strictly follow chronological order. We use $x^i_{a:b}$ to denote the concatenation of consecutive segments x^i_a through x^i_b. For any two alignments $(x^i_{a_s:a_e}, y^j_{b_s:b_e})$ and $(x^i_{c_s:c_e}, y^k_{d_s:d_e})$, Speech Vecalign guarantees that $i = j = k$ and that either $a_e < c_s, b_e < d_s$ or $a_s > c_e, b_s > d_e$. In contrast, Local Mining ensures $i = j = k$ but has no constraints on a, b, c, d, while Global Mining makes no guarantees at all.

3.3 Alignment Postprocessing

The goal of postprocessing is to clean the raw alignments and construct alignments with longer durations to improve S2ST models.

Removing low-quality alignments. First, we remove unaligned segments and high-cost alignments. The unaligned segments are due to deletions in the DP algorithm. Identical untranslated segments detected in Section 3.1 may fall into either category due to their 0-valued vectors.

Detection of identical untranslated segments, again. Occasionally, the location heuristic in Section 3.1 may fail, resulting in a small number of low-cost alignments with identical untranslated source and target segments. Searching is not needed at this step, as we already have the alignments. We apply Equation 6 to remaining alignments, where $\mathbf{A}$ and $\mathbf{B}$ are source and target segments in each alignment. We use the same thresholds in Section 3.1 to remove alignments.

Alignment concatenation. Another issue is that the raw alignments are too short: the average duration is 4.25 seconds, with 66% shorter than 5 seconds. To cover more context, we progressively concatenate each alignment with the subsequent ones. This can be easily done as alignments are in chronological order. Each concatenated alignment can contain up to 3 original alignments and span up to 20 seconds.

Global margin-scores computation. The raw alignments only have alignment costs as a quality indicator, which are computed *within* each document pair. To assess alignment quality *across* document pairs, we train FAISS (Johnson et al., 2019) indexes and compute margin-scores (Artetxe and Schwenk, 2019a) using Equation 1 for *all* obtained alignments, following the common strategy in MT dataset curation (Sloto et al., 2023).

Removing highly-overlapped alignments. Finally, we remove alignments that have too much overlap with others, following Duquenne et al. (2023a). For any two consecutive alignments, we compute the ratio of the overlapped source duration to the maximum duration of the two source segments. If the ratio exceeds a threshold, we discard the one with a lower margin-score. We train S2ST models with multiple threshold values to determine the best one. Our experiments in Appendix D.1 suggest that 0.4 work best for Global Mining and 0.8 work best for Local Mining and Speech Vecalign.

4 Experiments & Results

We apply Speech Vecalign, Global Mining, and Local Mining to the same raw data and train S2ST models on each type of alignments, providing a fair comparison.

4.1 Training Data

Data source. We use the unlabeled, unsegmented English and German plenary session recordings from VoxPopuli v1 (Wang et al., 2021a) as raw data. VoxPopuli contains European Parliament plenary session recordings in each of the 23 European Union languages, paired with spoken interpretations into the other languages. The document names are formatted as `${session_id}_${language}.ogg`, and paired documents have the same `${session_id}`. To avoid overlapping with the test set (Section 4.2), we only choose sessions from year 2013 to 2020. We also exclude sessions in the development set (Section 4.2). For En-to-De, the remaining data has 4,880 documents totaling about 3,000 hours for each language. For De-to-En, there are 5,782 documents totaling 3,400 hours per language. The difference is due to the different dev and test sets. All documents are in pairs, allowing all methods to have exactly the same raw data.

Speech Vecalign. We apply Speech Vecalign to each pair of speech documents and obtain alignments sorted by margin-scores. Training data is chosen in descending order of margin-scores. We train models on different data sizes and report the best results in Section 4.5. More details on data size optimization can be found in Appendix D.2.

Speech mining baselines. We apply Global Mining and Local Mining to the same raw data and embeddings as Speech Vecalign. The implementation is based on `stopes`[3] (Andrews et al., 2022). After mining, we apply the same postprocessing strategies in Section 3.3, except for alignment concatenation which is not applicable. Training data is chosen in descending order of margin-scores and details on data size optimization can be found in Appendix D.2.

4.2 Evaluation Data

Development set. Following Duquenne et al. (2023a), we choose 1000 samples from the highest scored sessions from the Voxpopuli S2ST dataset. Additionally, we avoid choosing sessions that occur on the same dates as the test set.

Test set. We use the Europarl-ST (EPST) test set (Iranzo-Sánchez et al., 2020) as an in-domain test set to evaluate the S2ST models. EPST is a multilingual S2TT dataset built on European Parliament debates from year 2008 to 2012. We also adopt FLEURS (Conneau et al., 2023) as an out-of-domain test set.

4.3 Experiment Setup

We train speech-to-unit translation (S2UT) models (Lee et al., 2022a) with `fairseq`[4] (Ott et al., 2019; Wang et al., 2020) on each type of alignments. The S2UT model takes source speech as input and predicts a sequence of target discrete units. The discrete units are obtained by applying a k-means model to the 11^{th} layer features of a HuBERT model (Hsu et al., 2021). For English, we use the mHuBERT from Lee et al. (2022b), and for German, we use the Germanic mHuBERT from

[3] https://github.com/facebookresearch/stopes
[4] https://github.com/facebookresearch/fairseq

Duquenne et al. (2023a). Consecutive duplicated units are removed. Our S2UT model architecture follows exactly Duquenne et al. (2023a). The architecture details and training hyperparameters are in Appendix B.

4.4 Evaluation Metrics

With the discrete units generated by S2UT models, we resynthesize speech using pretrained unit-based HiFi-GAN vocoders (Polyak et al., 2021) from Duquenne et al. (2023a). We then evaluate the resynthesized speech using both transcription-based and transcription-free methods.

For the transcription-based method, we transcribe the speech output using the same ASR models as Duquenne et al. (2023a). We evaluate the transcriptions using SacreBLEU[5] (Post, 2018) to compute BLEU[6] and chrF2++[7] scores. We apply the significance test using paired bootstrap resampling (Koehn, 2004) with 1000 bootstrap resamples.

We also adopt BLASER 2.0 (Dale and Costa-jussà, 2024) to directly evaluate speech output. We compute the referenced score using `blaser-2.0-ref`[8] for input and output speech, as well as the text reference. We compute the reference-free score using `blaser-2.0-qe`[9] for input and output speech only.

4.5 Main Results

As mentioned in Section 4.1, we train models on data of various sizes. Table 1 presents the best En-to-De and De-to-En results on the EPST test set, along with the corresponding data sizes. Additional results on the FLEURS test set are in Appendix E.

Intriguingly, for both directions, Speech Vecalign and speech mining models are competitive with or outperform SpeechMatrix (Duquenne et al., 2023a) models, despite the latter being mined from about 24k hours of speech per language, *8 times more* than our raw data.[10] For En-to-De, our Global Mining and Speech Vecalign models achieve improvements of 0.94 and 1.31 BLEU, respectively. Our Local Mining model achieves even 1.64 BLEU improvement. We suspect that SpeechMatrix has not removed identical untranslated segments prior to and after mining, which significantly hurts model performance. Further discussion is in Section 5.5.

While Local Mining has not been previously explored, our results suggest that it is a potentially useful method. Local Mining achieves the highest BLEU score in En-to-De, and only slightly underperforms Global Mining in De-to-En, indicating that constraining the mining scope to document pairs does not necessarily have a negative impact on alignment quality. Yet we note that Local Mining requires more training data to achieve its optimal performance, as shown in Appendix D.2.

Our Speech Vecalign models outperform both speech mining models in both directions. For En-to-De, the Speech Vecalign model achieves 12.58 BLEU, comparable with our strong Global Mining and Local Mining baselines. In terms of chrF2++, it surpasses Global Mining and Local Mining by 1.69 and 0.26, respectively. It also significantly improves their referenced BLASER 2.0 by 0.08 and 0.03. For De-to-En, Speech Vecalign and Global Mining models achieve comparable BLEU (16.14 vs. 15.96), but Speech Vecalign surpasses Global Mining by 0.57 in chrF2++. Speech Vecalign significantly outperforms Local Mining under all metrics. These results demonstrate that Speech Vecalign produces higher-quality alignments than both speech mining baselines.

5 Analysis

We analyze properties of speech mining methods and compare them with Speech Vecalign. Although we show that speech mining methods produce alignments similar to those of Speech Vecalign, the latter offers advantages of producing longer and less noisy alignments.

5.1 Speech Mining Mostly Locally Aligns Segments in Time Order

First, we show that Global Mining mostly **locally** aligns speech documents. While Global Mining searches for the best matching segment pairs among roughly 10 million segments, one might expect its alignments to cover the spread of the entire dataset. On the contrary, we find that Global Mining alignments are concentrated within docu-

[5] https://github.com/mjpost/sacrebleu

[6] Signature: nrefs:1 + case:mixed + eff:no + tok:13a + smooth:exp + version:2.2.0

[7] Signature: nrefs:1 + case:mixed + eff:yes + nc:6 + nw:2 + space:no + version:2.2.0

[8] https://huggingface.co/facebook/blaser-2.0-ref

[9] https://huggingface.co/facebook/blaser-2.0-qe

[10] We do not aim for state-of-the-art performance. Our results are not directly comparable to SpeechMatrix. We report SpeechMatrix results only to show the performance gap.

	Training Data		ASR-BLEU	ASR-chrF2++	BLASER 2.0	
	Alignment Method	# Hours			w/ text ref	w/o text ref
	English-to-German					
State-of-the-art	SpeechMatrix†	1451	10.1	-	-	-
	SpeechMatrix‡	1451	11.27	39.98	3.52	3.86
Baseline	Local Mining	1500	**12.91**	44.08	3.70	4.02
	Global Mining	1000	12.21	42.65	3.65	3.97
Our Method	Speech Vecalign	750	12.58	**44.34**	**3.73**	**4.05**
	p-value w.r.t Local Mining		0.1069	0.0999	0.0050*	0.0020*
	p-value w.r.t Global Mining		0.0769	0.0010*	0.0010*	0.0010*
	German-to-English					
State-of-the-art	SpeechMatrix†	1456	16.3	-	-	-
	SpeechMatrix‡	1456	16.62	43.77	3.81	4.11
Baseline	Local Mining	1250	15.64	42.85	3.70	4.02
	Global Mining	750	15.96	43.14	3.74	4.04
Our Method	Speech Vecalign	1000	**16.14**	**43.71**	**3.76**	**4.07**
	p-value w.r.t Local Mining		0.0030*	0.0010*	0.0010*	0.0010*
	p-value w.r.t Global Mining		0.1449	0.0030*	0.0030*	0.0010*

Table 1: Results for En-to-De and De-to-En on EPST test sets. **Bold** means better than speech mining baselines. Underline means the best overall. †Results from Duquenne et al. (2023a). ‡Models trained by ourselves. *p-value < 0.05. Results show that Speech Vecalign models perform better than baselines under almost all metrics in both directions.

ment pairs, each typically containing hundreds to thousands of segments.

To quantify this, we examine the 1000-hour Global Mining data and count alignments whose source and target segments come from *different* document pairs. As shown in Figure 3, fewer than 6% fall into this category, while the majority ($> 93\%$) are within paired documents.

Second, we analyze the time order of alignments produced by both speech mining methods. Borrowing the notation from Section 3.2, we define two pairs of alignments to be *in-order* if either $a_e < c_s, b_e < d_s$ or $a_s > c_e, b_s > d_e$; otherwise, they are *out-of-order*. Figure 3 shows that only around 1% alignments are out-of-order for both speech mining methods.

Observations above indicate that speech mining alignments are mostly within paired documents and preserve time order. We hypothesize that speech-to-speech alignments are sparse and high-quality ones mostly exist in paired documents.

As a by-product, this property can be leveraged to identify parallel documents. If Global Mining finds many alignments between two documents, they are likely to be parallel. It is particularly useful when the pairing metadata is not readily available.

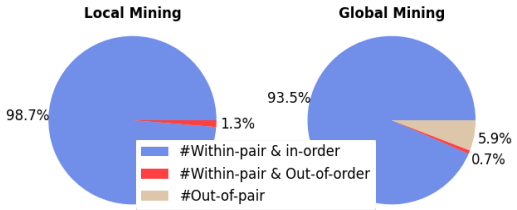


Figure 3: For both 1000-hour speech mining datasets, we compute percentages of En-De alignments that come from different document pairs, and those are within paired documents but out-of-order. Global Mining has 5.9% out-of-pair alignments, and both methods have only around 1% out-of-order alignments.

5.2 Speech Mining Methods Produce Similar Alignments as Speech Vecalign

Following the observations in Section 5.1 that speech mining produces mostly local, in-order alignments, we analyze the similarity between them and Speech Vecalign alignments.

We employ the alignment evaluation method[11] from Thompson and Koehn (2019), which computes precision and recall by comparing system alignments to a reference. There are two modes: *Strict*, which counts only exact matches as true positives, and *Lax*, which considers an alignment as true positive if both its source and target segment

[11] https://github.com/thompsonb/vecalign/blob/master/score.py.

overlap with the reference. If not true positive, an alignment is false positive. Recall is computed by swapping the reference and the system alignments.

Without loss of generality, we use Speech Vecalign En-De alignments as the reference, and evaluate speech mining ones. We choose 700k highest-scoring alignments from all three methods to ensure a fair comparison. Table 2 shows that about 30% of speech mining alignments are exactly the same as those of Speech Vecalign, and about 90% overlap with Speech Vecalign alignments. This high similarity explains why Speech Vecalign and speech mining models have similar performance.

Mode	Global Mining		Local Mining	
	Precision	Recall	Precision	Recall
Strict	0.325	0.326	0.305	0.305
Lax	0.865	0.965	0.963	0.814

Table 2: Precision and Recall for speech mining alignments when Speech Vecalign is used as the reference. The high precision and recall in the *Lax* mode indicate the methods produce similar alignments.

5.3 Speech Vecalign Produces Longer Alignments

As speech mining and Speech Vecalign produce similar alignments, we explore why Speech Vecalign models still perform better. A key advantage of Speech Vecalign is that it first produces fine-grained alignments and then constructs alignments with different amounts of context, thanks to the alignment concatenation strategy. Speech mining methods, on the other hand, solely depend on margin-scores and tend to favor shorter alignments.

With the best En-to-De models and corresponding data sizes from Section 4.5, Figure 4 presents the average sentence-level chrF2++ scores on the test set and the percentage of training alignments for different source speech duration ranges. Notably, Speech Vecalign has a large portion of long training samples: the blue bars are highest for durations longer than 12 seconds. Specifically, the average source duration of Speech Vecalign is 8.51 seconds, while Global Mining and Local Mining have average durations of 7.50 and 8.53 seconds, respectively. As a result, the Speech Vecalign model performs better on test samples longer than 10 seconds, while having comparable performance on shorter ones. This highlights that Speech Vecalign is able to produce longer, context-rich alignments which help to improve S2ST model performance. Interestingly, Local Mining surpasses the Global Mining model on long inputs, which could be also attributed to its longer training samples.

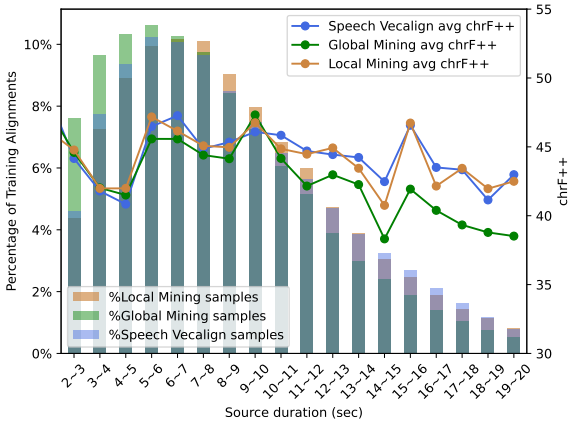


Figure 4: Average sentence-level chrF2++ on En-to-De test set and the corresponding portion of training samples with respect to duration ranges. The Speech Vecalign model consistently performs better on longer inputs.

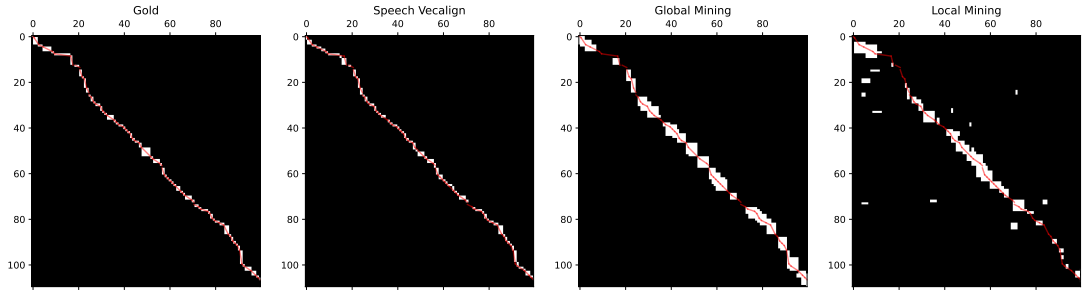
5.4 Speech Vecalign Produces Less Noise than Local Mining

We visualize alignments produced by different methods for the same document pair, which is about 10 minutes long and contains around 200 segments. For reference, we manually created a gold segment-level alignment, with detailed procedure in Appendix F. We illustrate the best 80 alignments for each of the speech mining methods.

As Figure 5 shows, Speech Vecalign produces the most fine-grained alignments and is most similar to the gold reference. Global Mining also performs well, aligning closely with the groundtruth path, whereas Local Mining produces more noise and misses more alignments along the correct path. We hypothesize Local Mining has limited number of segments in a single document pair, making nearest neighbors less effective normalizers in the margin-score computation.

5.5 Removing Identical Untranslated Segments is Critical

As presented in Section 4.5, our reproduced speech mining models achieve comparable or even better results than SpeechMatrix models. By listening to samples of SpeechMatrix alignments, we observed many cases where the source and target segments contained identical untranslated content, which is an issue mentioned in Section 3.1. Using the method described in Section 3.1, we identified



(a) Gold (b) Speech Vecalign (c) Global Mining (d) Local Mining

Figure 5: Visualizations of gold and 3 system alignments for 20180313-0900-PLENARY-15. The red lines indicate the gold alignment. Figures from left to right are: (a) Gold alignment manually created by us. (b) Raw Speech Vecalign alignments *without* alignment concatenation. (c) Global Mining. (d) Local Mining. Speech Vecalign and Global Mining follow closely with the gold alignment, while Local Mining produces more noise.

approximately 100k out of 630k alignments with untranslated source and target segments, totaling 181 hours.

To evaluate the impact of untranslated segments, we trained models on the original SpeechMatrix En-De alignments and on a version with untranslated alignments *removed*. The training data is chosen with a margin-score threshold of 1.09, following the original setup. As shown in Table 3, the cleaned data produces better models, improving BLEU score by **1.00** for En-to-De and **0.11** for De-to-En, despite having 13% less training data. The smaller gain on De-to-En may be due to most untranslated segments being in English, which have smaller impact on into-English translation.

We also re-produced our alignment pipelines *without* removing identical untranslated segments, referred to as "noisy" in Table 4. We trained models on 500 hours of this data. Although these untranslated segments account for less than 1% of the training data, performance degrades noticeably.

Overall, the experiments highlight that removing untranslated alignments is essential for S2ST training, corroborating Khayrallah and Koehn (2018), who found that the untranslated sentences are most catastrophic in neural machine translation.

Dataset	Hours	ASR-BLEU
English-to-German		
SpeechMatrix	1451	11.27
SpeechMatrix cleaned	1265	**12.27**
German-to-English		
SpeechMatrix	1456	16.62
SpeechMatrix cleaned	1276	**16.73**

Table 3: Performance of models trained on SpeechMatrix, before and after removing identical untranslated alignments. Results are measured on En-to-De and De-to-En EPST test sets. The removal of untranslated segments boosts model performance.

Dataset	#Noisy/All Aligns.	ASR-BLEU
Local Mining	-	**11.18**
Local Mining noisy	2.66k/236k	10.76
Global Mining	-	**11.54**
Global Mining noisy	1.46k/254k	11.27
Speech Vecalign	-	**11.78**
Speech Vecalign noisy	1.48k/222k	11.69

Table 4: Results for En-to-De on EPST test sets. "noisy" means the steps of removing identical untranslated segments are **not** applied. All datasets have 500 hours of training data.

6 Conclusion

We present Speech Vecalign, a parallel speech document alignment method that aligns speech segment embeddings within document pairs and in chronological order. We apply Speech Vecalign to parallel English-German VoxPopuli speech documents and conduct S2ST experiments to demonstrate its superiority over two strong speech mining baselines. Our analysis reveals that although speech mining methods primarily align documents locally and in-order, Global Mining falls short of producing long alignments, and Local Mining in particular produces more noise. For long-term future work, we plan to extend Speech Vecalign to other language pairs or other data sources. We can also explore aligning speech and text embeddings to construct S2TT datasets.

Limitations

Speech features for identical untranslated segment detection could be improved. Our current approach uses filterbank features, which are based on power spectrum, to detect identical untranslated

segments. However, filterbank features are likely to fail for segments that have identical content but differ in signal power. As one of the anonymous reviewers pointed out, cepstral features may be a more robust alternative.

Limited language pair. We have only conducted experiments for English and German speech from the VoxPopuli dataset. As Speech Vecalign heavily relies on the quality of speech embeddings, the performance is unclear for other language pairs and other domains of speech.

Dependency on parallel speech documents. Speech Vecalign requires parallel speech documents, which is often not available. We may rely on Global Mining to discover parallel documents, as Section 5.1 suggests, but doing so will introduce extra computation costs.

Acknowledgments

This work was in part supported by a Sony Research Award. We thank Antonios Anastasopoulos for proofreading an early version of this paper. We are thankful to anonymous reviewers for their valuable feedback.

References

Pierre Andrews, Guillaume Wenzek, Kevin Heffernan, Onur Çelebi, Anna Sun, Ammar Kamran, Yingzhe Guo, Alexandre Mourachko, Holger Schwenk, and Angela Fan. 2022. stopes - modular machine translation pipelines. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 258–265, Abu Dhabi, UAE. Association for Computational Linguistics.

Mikel Artetxe and Holger Schwenk. 2019a. Margin-based parallel corpus mining with multilingual sentence embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3197–3203, Florence, Italy. Association for Computational Linguistics.

Mikel Artetxe and Holger Schwenk. 2019b. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Richard Bellman. 1954. The theory of dynamic programming. *Bulletin of the American Mathematical Society*, 60(6):503–515.

Peter F. Brown, Jennifer C. Lai, and Robert L. Mercer. 1991. Aligning sentences in parallel corpora. In *29th Annual Meeting of the Association for Computational Linguistics*, pages 169–176, Berkeley, California, USA. Association for Computational Linguistics.

Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna. 2023. Fleurs: Few-shot learning evaluation of universal representations of speech. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 798–805.

David Dale and Marta R. Costa-jussà. 2024. BLASER 2.0: a metric for evaluation and quality estimation of massively multilingual speech and text translation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16075–16085, Miami, Florida, USA. Association for Computational Linguistics.

Paul-Ambroise Duquenne, Hongyu Gong, Ning Dong, Jingfei Du, Ann Lee, Vedanuj Goswami, Changhan Wang, Juan Pino, Benoît Sagot, and Holger Schwenk. 2023a. SpeechMatrix: A large-scale mined corpus of multilingual speech-to-speech translations. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16251–16269, Toronto, Canada. Association for Computational Linguistics.

Paul-Ambroise Duquenne, Hongyu Gong, and Holger Schwenk. 2021. Multimodal and Multilingual Embeddings for Large-Scale Speech Mining. In *Advances in Neural Information Processing Systems*, volume 34, pages 15748–15761. Curran Associates, Inc.

Paul-Ambroise Duquenne, Holger Schwenk, and Benoit Sagot. 2023b. SONAR: sentence-level multimodal and language-agnostic representations. *arXiv preprint*.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

William A. Gale and Kenneth W. Church. 1993. A program for aligning sentences in bilingual corpora. *Computational Linguistics*, 19(1):75–102.

Luís Gomes and Gabriel Pereira Lopes. 2016. First steps towards coverage-based sentence alignment. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2228–2231, Portorož, Slovenia. European Language Resources Association (ELRA).

Kevin Heffernan, Onur Çelebi, and Holger Schwenk. 2022. Bitext mining using distilled sentence representations for low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2101–2112, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Kevin Heffernan, Artyom Kozhevnikov, Loic Barrault, Alexandre Mourachko, and Holger Schwenk. 2024.

Aligning speech segments beyond pure semantics. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3626–3635, Bangkok, Thailand. Association for Computational Linguistics.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460.

Hirofumi Inaguma, Sravya Popuri, Ilia Kulikov, Peng-Jen Chen, Changhan Wang, Yu-An Chung, Yun Tang, Ann Lee, Shinji Watanabe, and Juan Pino. 2023. UnitY: Two-pass direct speech-to-speech translation with discrete units. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15655–15680, Toronto, Canada. Association for Computational Linguistics.

Javier Iranzo-Sánchez, Joan Albert Silvestre-Cerdà, Javier Jorge, Nahuel Roselló, Adrià Giménez, Albert Sanchis, Jorge Civera, and Alfons Juan. 2020. Europarl-ST: A Multilingual Corpus for Speech Translation of Parliamentary Debates. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8229–8233.

Ye Jia, Michelle Tadmor Ramanovich, Tal Remez, and Roi Pomerantz. 2022a. Translatotron 2: High-quality direct speech-to-speech translation with voice preservation. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 10120–10134. PMLR.

Ye Jia, Michelle Tadmor Ramanovich, Quan Wang, and Heiga Zen. 2022b. CVSS corpus and massively multilingual speech-to-speech translation. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 6691–6703, Marseille, France. European Language Resources Association.

Ye Jia, Ron J. Weiss, Fadi Biadsy, Wolfgang Macherey, Melvin Johnson, Zhifeng Chen, and Yonghui Wu. 2019. Direct Speech-to-Speech Translation with a Sequence-to-Sequence Model. In *Proc. Interspeech 2019*, pages 1123–1127.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547.

Huda Khayrallah and Philipp Koehn. 2018. On the impact of various types of noise on neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 74–83, Melbourne, Australia. 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. 2017. Adam: A method for stochastic optimization. *Preprint*, arXiv:1412.6980.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain. Association for Computational Linguistics.

Alon Lavie, A. Waibel, Lori Levin, M. Finke, Donna Gates, Marsal Gavalda, Torsten Zeppenfeld, and Puming Zhan. 1997. Janus-III: speech-to-speech translation in multiple languages. pages 99 – 102 vol.1.

Ann Lee, Peng-Jen Chen, Changhan Wang, Jiatao Gu, Sravya Popuri, Xutai Ma, Adam Polyak, Yossi Adi, Qing He, Yun Tang, Juan Pino, and Wei-Ning Hsu. 2022a. Direct speech-to-speech translation with discrete units. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3327–3339, Dublin, Ireland. Association for Computational Linguistics.

Ann Lee, Hongyu Gong, Paul-Ambroise Duquenne, Holger Schwenk, Peng-Jen Chen, Changhan Wang, Sravya Popuri, Yossi Adi, Juan Pino, Jiatao Gu, and Wei-Ning Hsu. 2022b. Textless speech-to-speech translation on real data. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 860–872, Seattle, United States. Association for Computational Linguistics.

Robert C. Moore. 2002. Fast and accurate sentence alignment of bilingual corpora. In *Proceedings of the 5th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 135–144, Tiburon, USA. Springer.

S. Nakamura, K. Markov, H. Nakaiwa, G. Kikui, H. Kawai, T. Jitsuhiro, J.-S. Zhang, H. Yamamoto, E. Sumita, and S. Yamamoto. 2006. The ATR Multilingual Speech-to-Speech Translation System. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(2):365–376.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Adam Polyak, Yossi Adi, Jade Copet, Eugene Kharitonov, Kushal Lakhotia, Wei-Ning Hsu, Abdelrahman Mohamed, and Emmanuel Dupoux. 2021. Speech Resynthesis from Discrete Disentangled Self-Supervised Representations. In *Proc. Interspeech 2021*, pages 3615–3619.

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.

Matt Post, Gaurav Kumar, Adam Lopez, Damianos Karakos, Chris Callison-Burch, and Sanjeev Khudanpur. 2013. Improved speech-to-text translation with the fisher and callhome Spanish-English speech translation corpus. In *Proceedings of the 10th International Workshop on Spoken Language Translation: Papers*, Heidelberg, Germany.

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*, volume 202 of *Proceedings of Machine Learning Research*, pages 28492–28518. PMLR.

Stan Salvador and Philip Chan. 2007. Toward accurate dynamic time warping in linear time and space. *Intelligent Data Analysis*, 11(5):561–580.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, Armand Joulin, and Angela Fan. 2021. CCMatrix: Mining billions of high-quality parallel sentences on the web. 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 6490–6500, Online. Association for Computational Linguistics.

Seamless Communication, Loïc Barrault, Yu-An Chung, Mariano Coria Meglioli, David Dale, Ning Dong, Mark Duppenthaler, Paul-Ambroise Duquenne, Brian Ellis, Hady Elsahar, Justin Haaheim, John Hoffman, Min-Jae Hwang, Hirofumi Inaguma, Christopher Klaiber, Ilia Kulikov, Pengwei Li, Daniel Licht, Jean Maillard, and 46 others. 2023. Seamless: Multilingual Expressive and Streaming Speech Translation. *Preprint*, arXiv:2312.05187.

Rico Sennrich and Martin Volk. 2010. MT-based sentence alignment for OCR-generated parallel texts. In *Proceedings of the 9th Conference of the Association for Machine Translation in the Americas: Research Papers*, Denver, Colorado, USA. Association for Machine Translation in the Americas.

Silero Team. 2021. Silero VAD: pre-trained enterprise-grade Voice Activity Detector (VAD), Number Detector and Language Classifier. `https://github.com/snakers4/silero-vad`.

Steve Sloto, Brian Thompson, Huda Khayrallah, Tobias Domhan, Thamme Gowda, and Philipp Koehn. 2023. Findings of the WMT 2023 shared task on parallel data curation. In *Proceedings of the Eighth Conference on Machine Translation*, pages 95–102, Singapore. Association for Computational Linguistics.

Brian Thompson and Philipp Koehn. 2019. Vecalign: Improved sentence alignment in linear time and space. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1342–1348, Hong Kong, China. Association for Computational Linguistics.

Dániel Varga, Péter Halácsy, András Kornai, Viktor Nagy, László Németh, and Viktor Trón. 2007. Parallel corpora for medium density languages. *Amsterdam Studies In The Theory And History Of Linguistic Science Series 4*, 292:247.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Wolfgang Wahlster. 2013. *Verbmobil: foundations of speech-to-speech translation*. Springer Science & Business Media.

Changhan Wang, Morgane Riviere, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson, Juan Pino, and Emmanuel Dupoux. 2021a. VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation. 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 993–1003, Online. Association for Computational Linguistics.

Changhan Wang, Yun Tang, Xutai Ma, Anne Wu, Dmytro Okhonko, and Juan Pino. 2020. Fairseq S2T: Fast speech-to-text modeling with fairseq. In *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*, pages 33–39, Suzhou, China. Association for Computational Linguistics.

Changhan Wang, Anne Wu, Jiatao Gu, and Juan Pino. 2021b. CoVoST 2 and Massively Multilingual Speech Translation. In *Proc. Interspeech 2021*, pages 2247–2251.

A Related Work

Speech-to-speech translation (S2ST). The early S2ST systems consist of cascaded ASR, MT, and TTS models (Lavie et al., 1997; Nakamura et al.,

2006; Wahlster, 2013). Direct S2ST models have recently been proposed to alleviate error propagation, support unwritten languages, and improve inference speed. Translatotron models (Jia et al., 2019, 2022a) are trained with spectrograms as targets, while the S2UT model (Lee et al., 2022a) outputs discrete units. UnitY (Inaguma et al., 2023) and UnitY2 (Seamless Communication et al., 2023) are two-pass direct S2ST models that predict both subwords and discrete units with a single model. Despite advances in architectures, the amount of supervised training data is still insufficient and thus limits model performance.

Bilingual text sentence alignment. Text alignment is very related to speech alignment. Methods apply dynamic programming (Bellman, 1954) and mainly differ in the design of scoring functions. Early works (Brown et al., 1991; Gale and Church, 1993) are based on sentence lengths. Later methods incorporate translations in various ways (Moore, 2002; Varga et al., 2007; Sennrich and Volk, 2010; Gomes and Lopes, 2016). Our work is inspired by Vecalign (Thompson and Koehn, 2019), which utilizes margin-based cosine similarities between multilingual sentence embeddings like LASER (Artetxe and Schwenk, 2019b; Heffernan et al., 2022) and LaBSE (Feng et al., 2022). Vecalign is also more efficient than previous methods. By applying fast dynamic time warping (Salvador and Chan, 2007), it has a linear time and space complexity with respect to the number of input sentences. The recent progress of extending multilingual sentence embeddings to the speech modality (Duquenne et al., 2021; Khurana et al., 2022; Duquenne et al., 2023a,b) enables us to align speech segments by their speech embeddings using the same algorithm.

S2ST datasets. There are two common ways to automatically build an S2ST dataset: (1) building alignments from multilingual speech data; (2) synthesizing speech for text translations from existing speech-to-text translation (S2TT) corpora. The first line of work has human spoken speech on both source and target sides. VoxPopuli (Wang et al., 2021a) aligns multilingual speech documents based on text transcriptions, yielding 17.3k-hour alignments between 15 source and target languages. SpeechMatrix (Duquenne et al., 2023a) applies Global Mining with SpeechLASER embeddings on VoxPopuli. It obtains alignments for 136 language pairs with an average of 1537 hours per direction. Seamless Communication et al. (2023) apply Global Mining to web-crawled speech data with SONAR embeddings. Seamless Communication et al. (2023) also mine a SeamlessAlignExpressive dataset with expressively- and semantically-aligned segment pairs, based on a blend of both semantic and prosodic similarity score (Heffernan et al., 2024).

The second line of work has synthesized speech on the target side. Fisher (Post et al., 2013) is a Spanish-English S2TT dataset containing about 170 hours of Spanish telephone conversations and English translations which are used to synthesize English speech. CVSS (Jia et al., 2022b) is an S2ST dataset covering utterances from 21 languages to English, obtained by synthesizing the text translations in CoVoST 2 (Wang et al., 2021b). Besides automatic methods, FLEURS (Conneau et al., 2023) has collected human read speech covering 102 languages. But it contains only about 12 hours per language and is intended for evaluation.

B Speech-to-Speech Translation

We describe the model architecture in Appendix B.1 and the experiment hyperparameters in Appendix B.2.

B.1 S2UT Model Architecture

The S2UT model (Lee et al., 2022a) adopts the Transformer encoder-decoder architecture (Vaswani et al., 2017). It has 2 convolutional layers, 12 transformer encoders, and 6 transformer decoders for target unit prediction. Additionally, 2 transformer decoders for source unit prediction are attached to the 6^{th} encoder layer. All embedding dimensions are 512, except for the source unit decoder, which has a dimension of 256. The forward dimensions are 2048. The model has a total of 70M trainable parameters.

B.2 Training Hyperparameters

We train the models using a learning rate of 0.0005 with the inverse_sqrt scheduler. We use the adam optimizer (Kingma and Ba, 2017) with betas $(0.9, 0.98)$. We apply a dropout rate of 0.3 and a label smoothing weight of 0.2.

Due to limited computing resources, we adopt different training strategies for different purposes. The 500-hour datasets are used for hyperparameter optimization, and larger datasets are used for reporting main results. All models are trained for up to 400k steps, with the first 10,000 steps as

a warmup stage. For experiments on a 500-hour dataset, we use a batch size of 320k tokens and apply early-stopping if there is no improvement on the development set for 30 epochs. These models are trained on 4 NVIDIA GeForce GTX 1080 Ti GPUs for approximately 15 days. For larger datasets, we increase the batch size to 640k tokens and early-stopping is not applied. These models are trained on 2 NVIDIA A100-SXM4-80GB GPUs for approximately 15 days. The best checkpoint is selected based on the development set loss. All experiments are conducted in fp32, as we found training with fp16 and amp very unstable.

C Computation Costs for Alignment

Segment embedding. This is the most time-consuming step. We use a mixture of NVIDIA GeForce GTX 1080 and 2080 Ti GPUs. Embedding about 6,000 hours of speech (3,000 hours for each language) took approximately 1,100 GPU hours.

Alignment. Local Mining and Global Mining run on a single GPU. They take about 2 hours. Speech Vecalign runs on a single CPU and takes about 2 hours.

D Training Data Optimization

There are two hyperparameters that affect training data: (1) the maximum source duration overlap ratio between alignments, $max\_overlap$, which is mentioned in Section 3.3, and (2) the data size.

$max\_overlap$ controls the trade-off between overlapped durations and data quality. For instance, a lower $max\_overlap$ reduces the overlap but also discards alignments more aggressively. Overlapped alignments usually have similar margin-scores, so more high-quality alignments are lost. The data size controls the trade-off between data size and data quality cutoff. For instance, a larger dataset will have a lower quality cutoff, as alignments are selected in descending order of margin-scores. In this section, we optimize the combination of $max\_overlap$ and data size by training S2UT models on different datasets. Note that the raw data stays the same.

D.1 Optimizing $max\_overlap$

We first experiment with different values of $max\_overlap$. We apply different $max\_overlap$ thresholds during the postprocessing stage, and always choose the best 500 hours as the training data. The optimal value is determined based on development set ASR-BLEU. Figure 6 shows that 0.8 works best for Speech Vecalign and Local Mining and 0.4 works best for Global Mining. The test set performance is also drawn in Figure 6, exhibiting a similar trend.

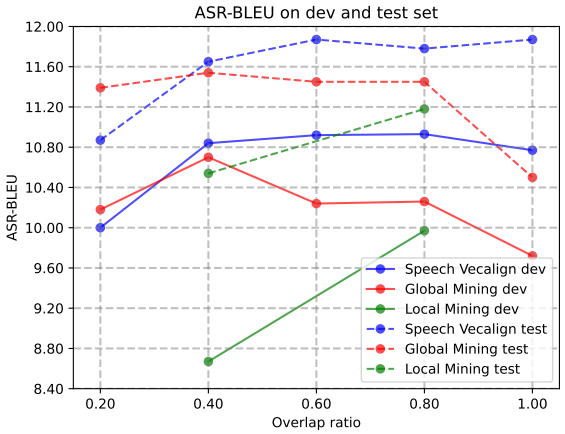


Figure 6: ASR-BLEU on En-to-De EPST dev and test set. All models are trained with 500-hour data. Only $max\_overlap$ varies.

D.2 Optimizing Training Data Size

Next we optimize the training data size. We fix $max\_overlap$ at 0.4 for Global Mining and 0.8 for Speech Vecalign and Local Mining during postprocessing, only lowering the quality cutoff to include more training data. The models are trained on different amounts of data until we find the peak performance. Results are shown in Figure 7.

For En-to-De, the best Speech Vecalign model is trained on the 750-hour dataset, achieving 12.58 BLEU. It outperforms the best Global Mining model which achieves 12.21 BLEU. The best Local Mining model achieves 12.91 BLEU. However, we note that it requires a lot more data than the other two methods to achieve the peak performance.

For De-to-En, the 1000-hour dataset works best for Speech Vecalign while the 750-hour dataset works best for Global Mining. Local Mining achieves the peak performance when the data size is 1250 hours, still requiring more data than the other methods. The Speech Vecalign performs better than both the Global Mining and the Local Mining models.

E Evaluation Results on FLEURS

We provide evaluation results on the FLEURS test set in Table 5. Similar to Section 4.5, our results match or outperform SpeechMatrix results. For

	Training Data		ASR-BLEU	ASR-chrF2++	BLASER 2.0	
	Alignment Method	# Hours			w/ text ref	w/o text ref
	English-to-German					
State-of-the-art	SpeechMatrix†	1451	2.7	-	-	-
	SpeechMatrix‡	1451	3.36	25.65	2.50	2.87
Baseline	Local Mining	1500	**3.93**	**28.86**	2.53	2.91
	Global Mining	1000	3.42	27.84	2.49	2.86
Our Method	Speech Vecalign	750	3.73	28.69	**2.55**	**2.94**
	p-value w.r.t Local Mining		0.1638	0.1938	0.0659	0.0610
	p-value w.r.t Global Mining		0.0939	0.0050*	0.0020*	0.0010*
	German-to-English					
State-of-the-art	SpeechMatrix†	1456	8.3	-	-	-
	SpeechMatrix‡	1456	12.18	35.08	3.16	3.51
Baseline	Local Mining	1250	11.52	37.88	3.09	3.46
	Global Mining	750	**11.55**	38.24	3.12	3.49
Our Method	Speech Vecalign	1000	11.39	**38.34**	**3.16**	**3.53**
	p-value w.r.t Local Mining		0.2468	0.0829	0.0010*	0.0010*
	p-value w.r.t Global Mining		0.2907	0.2118	0.0160*	0.0120*

Table 5: Results for En-to-De and De-to-En on FLEURS test sets. **Bold** means better than speech mining baselines. Underline means the best overall. †Results from Duquenne et al. (2023a). ‡Models trained by ourselves. *p-value < 0.05. Results show that Speech Vecalign models perform better than baselines under most metrics in both directions.

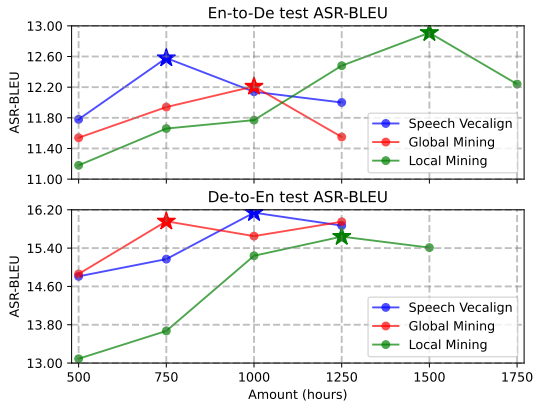


Figure 7: ASR-BLEU on En-to-De (above) and De-to-En (below) EPST test set. Models are trained on different amounts of training data. The best Speech Vecalign models outperform the other speech mining models.

both En-to-De and De-to-En, Speech Vecalign and Global Mining achieve comparable performance when using the transcription-based metrics ASR-BLEU and ASR-chrF2++. Their performance is especially close on De-to-En. However, Speech Vecalign is significantly better than Global Mining when using the BLASER 2.0 metrics, achieving an improvement of 0.06 and 0.04 referenced BLASER 2.0 scores on En-to-De and De-to-En, respectively.

For En-to-De, Speech Vecalign achieves comparable performance with Local Mining on all metrics. For De-to-En, Speech Vecalign is significantly better than Local Mining when using BLASER 2.0 metrics.

Overall, we can show that Speech Vecalign performs better than both Local Mining and Global Mining.

F Procedure of Manual Alignments

The manual alignment procedure is as follows:

1. We apply Whisper (Radford et al., 2023) to transcribe the German and English speech documents;

2. We manually select the corresponding words for each speech segment from the obtained transcriptions;

3. We used Google Translate to translate the German transcriptions into English, as the author does not speak German;

4. We align the German and English segments based on the corresponding English transcriptions and translations.

Although this process depends on models such as Whisper and Google Translate, we argue that they

should perform extremely well on German and English and should produce almost perfect transcriptions and translations.

G Evaluation of System Alignments using the Manual Alignments

We use the same alignment evaluation method as Section 5.2, but we use the manual alignments as the reference. There are 144 raw Speech Vecalign alignments, and we choose the same number of alignments from Global Mining and Local Mining in descending order of margin-scores. The Recall and Precision of raw Speech Vecalign, Local Mining, and Global Mining alignments are presented in Table 6.

The three methods have similar Lax Precisions, while that of Local Mining and Global Mining are slightly higher. Speech Vecalign has the highest recall values than both the speech mining baselines. Among the three methods, Local Mining has the worst performance in general. This follows Figure 5 that both Speech Vecalign and Global Mining have good performance but Local Mining does not perform well.

	Precision		**Recall**	
	Strict	*Lax*	*Strict*	*Lax*
raw Local Mining	0.139	**0.993**	0.147	0.676
Global Mining	0.188	**0.993**	0.199	0.868
raw Speech Vecalign	**0.597**	0.979	**0.632**	**0.978**

Table 6: Precision and Recall for raw Speech Vecalign, Global Mining and raw Local Mining alignments when manual alignments are used as the reference.

H Statistics for Intermediate Procedures

As our proposed alignment pipeline consists of several intermediate steps, we report numbers of segments or alignments in Table 7. We use English-to-German alignment as an example.

Stage	# En Segments	# De Segments	# Alignments
Segmentation	4,113,319	3,056,797	-
Detection of identical untranslated segments	47,008	47,008	-
Segment concatenation	19,331,307	13,012,502	-
Vecalign	-	-	2,597,796
After removing low-quality alignments	-	-	1,968,141
After removing untranslated alignments	-	-	1,962,032
Alignment concatenation	-	-	3,661,860
After removing short alignments ($<$ 1 second)	-	-	2,810,697
After removing highly-overlapped alignments ($max\_overlap < 0.8$ and duration $<$ 2 seconds)	-	-	851,446
Choose the best 750 hours	-	-	317,293

Table 7: Number of segments or alignments at each stage.