WHO TAUGHT THE MODEL TO SIGN? FINE-GRAINED DATA ATTRIBUTION FOR OPTIMIZING SIGN LANGUAGE PRODUCTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Gloss-to-Pose (G2P) generation translates a discrete gloss sequence into a continuous sign language pose sequence, where each gloss represents a basic semantic unit. Recent generative models have improved overall G2P quality, yet a model may still generate poor poses for only a few glosses while producing the rest of the sequence well. However, existing methods mainly rely on sequence-level supervision, making it difficult to provide targeted supervision for an individual gloss. We study this problem from the perspective of training data. We refer to each occurrence of a gloss in a training sample as a *training occurrence*. Different occurrences of the same gloss may provide different supervision due to variations in sentence context and pose quality. This raises a key question: which training occurrences provide more useful supervision for generating a particular gloss, and how can this information be used to improve G2P training? To address this problem, we introduce SignTrak, an occurrence-level data attribution and optimization framework for G2P. First, to enable occurrence-level attribution under sequence-level supervision, SignTrak uses gradient sensitivity to allocate temporal generation losses to individual gloss occurrences, yielding occurrence-level losses without requiring explicit pose segmentation. Based on these losses, we apply gradient-based data attribution to estimate how each training occurrence influences each target occurrence of the same gloss. Second, to use the estimated occurrence-level influence to improve G2P training, we introduce attribution-guided optimization, which aggregates each training occurrence's influence over multiple target occurrences of the same gloss and converts the aggregated influence into its training weight for fine-tuning. Extensive experiments demonstrate improvements in both local pose generation and overall G2P performance.

1 INTRODUCTION

Gloss-to-Pose (G2P) (Saunders et al., 2020b; Xie et al., 2024) generation aims to translate a discrete gloss sequence into a continuous sign language pose sequence, where a gloss denotes a basic semantic unit used to represent a sign. Recent advances in generative models, including Transformers (Saunders et al., 2021b), diffusion models (Yu et al., 2026; Tang et al., 2025a), have improved the overall quality of G2P generation. Nevertheless, a model can still generate poor poses for only a few glosses while generating the rest of the sequence well, as illustrated in Figure 1(a). However, existing methods mainly rely on sequence-level supervision, making it difficult to provide targeted supervision for an individual gloss. In this work, we study this problem from the perspective of training data. Each training sample consists of a gloss sequence paired with its corresponding pose sequence. We refer to each occurrence of a gloss in a training sample as a *training gloss occurrence*, or simply a *training occurrence*. This variation can arise from both sentence context, which affects signing poses through transitions and co-articulation (Saunders et al., 2022), and differences in the quality of the corresponding pose data across occurrences. As illustrated in Figure 1(b), this leads to the central question of this work: **which training occurrences provide more useful supervision for a target gloss occurrence in its sentence context, and how can this information be used to improve G2P training?**

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

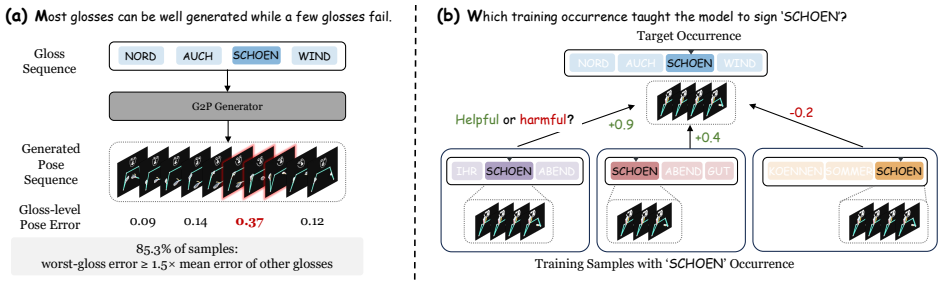


Figure 1: Motivation. (a) A generated pose sequence may be accurate for most glosses while failing on a few (with gloss-aligned pose segments obtained via CTC). (b) Since different training occurrences of the same gloss may contribute differently to generating a target occurrence, we ask: which training occurrences taught the model how to generate this occurrence?

Data attribution (Koh & Liang, 2017) provides a natural way to study this question by estimating how a training sample influences the model's prediction on a target sample. However, existing data attribution methods generally treat a complete training sample as the unit of attribution (Park et al., 2023). Sample-level attribution estimates the influence of a complete training sequence on a target sample, but cannot separate the contribution of individual gloss occurrences within that sequence. This limitation arises because the sequence-level training loss jointly reflects supervision from multiple gloss occurrences in the sample. It therefore cannot determine how an individual training occurrence affects the generation of a target occurrence. Estimating influence between a training occurrence and a target occurrence requires decomposing the sequence-level loss into occurrence-level losses. A straightforward solution is to divide the pose sequence into temporal segments and compute the loss within the segment aligned with each gloss. However, explicit gloss-level pose boundaries are not available under sequence-level supervision, and estimated boundaries can be unreliable because pose transitions and co-articulation between neighboring glosses often span these boundaries. The first challenge is therefore **how to derive occurrence-level losses from sequence-level supervision without relying on explicit pose segmentation, and use them to estimate the influence of training occurrences on target occurrences of the same gloss**.

However, the influence between a training occurrence and a single target occurrence is not sufficient to determine how strongly the training occurrence should be weighted during training. Because the same gloss can appear in different sentence contexts, the influence of a training occurrence may vary across target occurrences of that gloss. We therefore need to aggregate the influence of a training occurrence over multiple target occurrences before determining its training weight. Moreover, each training sample contains multiple gloss occurrences. Reweighting the whole training sample based on one occurrence would also change the supervision of the other occurrences in that sample. The second challenge is therefore **how to aggregate the influence of a training occurrence across multiple target occurrences and convert the aggregated influence into an occurrence-specific training weight**.

To address these challenges, we propose **SignTrak**, an occurrence-level data attribution and optimization framework for G2P. First, we introduce *occurrence-level attribution*. To enable attribution at occurrence level, we use gradient sensitivity to decompose the sequence-level generation loss into occurrence-level losses, without imposing explicit gloss-level pose boundaries. To obtain target occurrences for attribution, we construct a separate *query set*; from this point onward, we refer to target gloss occurrences drawn from this set as *query occurrences*. Based on the occurrence-level losses, we apply gradient-based data attribution to estimate the influence of each training occurrence on individual query occurrences of the same gloss. Second, we introduce *attribution-guided optimization*, which converts the estimated influence into training weights. For each training occurrence, we aggregate its influence over multiple query occurrences of the same gloss and use the aggregated influence to adjust the weight of its occurrence-level loss. Reweighting is performed among training occurrences of the same gloss, allowing the model to emphasize occurrences that provide more useful supervision while avoiding direct reweighting of other glosses in the same training sample. Our main contributions are summarized as follows:

- To investigate which training data contribute more to the generation of a particular gloss, we introduce **occurrence-level data attribution for G2P**, which estimates the influence of individual training occurrences on target occurrences of the same gloss.

- To enable occurrence-level attribution under sequence-level supervision, we propose a **sensitivity-guided loss decomposition** that allocates temporal losses to gloss occurrences according to their gradient sensitivity, without relying on explicit pose segmentation.
- To use the estimated occurrence-level influence to improve G2P training, we introduce **attribution-guided optimization**, which aggregates each training occurrence's influence over multiple target occurrences of the same gloss and converts the aggregated influence into an occurrence-specific training weight. Extensive experiments demonstrate improvements in both local pose generation and overall G2P performance.

2 BACKGROUND

2.1 FLOW MATCHING FOR GLOSS TO POSE GENERATION

Let $\mathcal{D} = \{z_n\}_{n=1}^{N}$ denote a G2P task training set, where each sample $z_n = (G_n, P_n)$ consists of a discrete gloss sequence $G_n = (g_{n,1}, \ldots, g_{n,K_n})$ and a continuous pose sequence $P_n \in \mathbb{R}^{T_n \times d_p}$. A frozen variational autoencoder (VAE) encodes P into a latent sequence $H \in \mathbb{R}^{L \times d_h}$, on which conditional flow matching (Lipman et al., 2022) (FM) is performed. Given a latent sequence H and Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$, we construct a linear probability path as

$$x_t = (1-t)\epsilon + tH, \qquad \dot{x}_t = H - \epsilon, \quad t \in [0,1]. \tag{1}$$

Here, t denotes the flow time, distinct from the latent temporal index. A gloss-conditioned vector field $v_\theta(x_t, t, G)$ predicts the latent velocity and is trained with

$$\mathcal{L}_{\mathrm{FM}}(z;\theta) = \mathbb{E}_{t,\epsilon}\left[\frac{1}{Ld_h}\left\|v_\theta(x_t,t,G) - \dot{x}_t\right\|_F^2\right]. \tag{2}$$

The FM objective can be equivalently decomposed over latent temporal positions as

$$\mathcal{L}_{\mathrm{FM}}(z;\theta) = \mathbb{E}_{t,\epsilon}\left[\sum_{m=1}^{L} \ell_m(z;\theta,H,t,\epsilon)\right]. \tag{3}$$

Here, $\ell_m(z;\theta,H,t,\epsilon) = \frac{1}{Ld_h}\|[v_\theta(x_t,t,G)]_m - [\dot{x}_t]_m\|_2^2$ denotes the loss contribution at latent temporal position m. Thus, although the flow-matching objective contains temporally localized prediction errors, these errors are aggregated into a sequence-level objective and are not explicitly assigned to individual gloss occurrences.

2.2 DATA ATTRIBUTION

Data attribution estimates the contribution of individual training samples to model behavior on a query sample (Koh & Liang, 2017). For a training sample $z_n \in \mathcal{D}$ and a query sample z_q, let θ denote the learned model parameters and $\mathcal{L}(z;\theta)$ denote the sample loss. Influence functions pproximate the effect of infinitesimally upweighting z_n on the query loss as

$$\mathcal{I}(z_n, z_q) = -\nabla_\theta \mathcal{L}(z_q;\theta)^\top H_\theta^{-1} \nabla_\theta \mathcal{L}(z_n;\theta), \tag{4}$$

where $H_\theta = \frac{1}{N}\sum_{n=1}^{N} \nabla_\theta^2 \mathcal{L}(z_n;\theta)$ is the Hessian of the empirical training objective. However, the high dimensionality of modern neural networks makes explicit Hessian construction and inversion computationally prohibitive. TRAK (Park et al., 2023) improves scalability by representing each sample with a low dimensional projected gradient. Given a measurement function $F(z;\theta)$ and a random projection matrix $P \in \mathbb{R}^{d \times k}$ with $k \ll d$, the gradient feature is $\phi(z) = P^\top \nabla_\theta F(z;\theta)$. TRAK combines these projected gradients with a low-dimensional feature-space preconditioner, avoiding inversion in the full parameter space. Recent work has extended gradient based attribution to generative models. D-TRAK (Zheng et al., 2024) shows that attribution quality is sensitive to the model output function used to construct gradient features, and that the standard diffusion training loss can be suboptimal for attribution. More recently, Concept-TRAK (Park et al., 2025) extends diffusion attribution to concept-specific influence, while MoTIVE (Wu et al., 2026) introduces motion-specific attribution signals for video generation to emphasize dynamic information over static appearance. Despite targeting more specific generative behaviors, these methods still assign attribution scores to whole training examples. In G2P, however, each training sequence contains multiple gloss occurrences, and the attribution unit is an individual occurrence within the sequence.

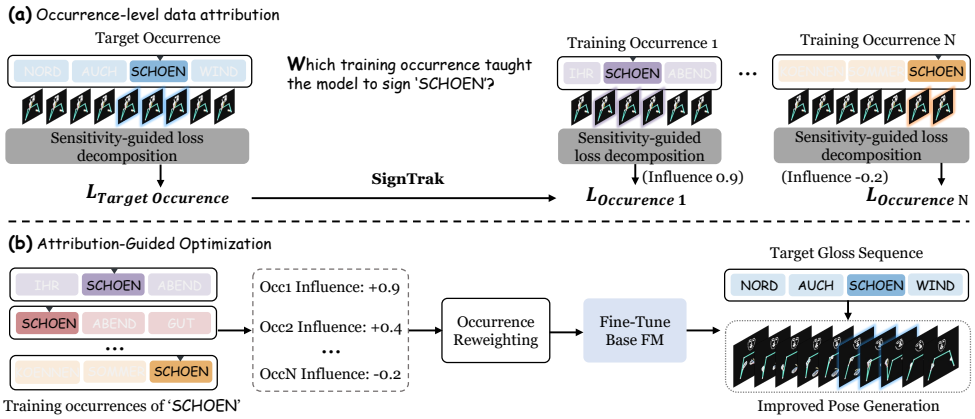


Figure 2: Overview of SignTrak.

3 METHOD

We present SignTrak, an occurrence-level data attribution and optimization framework for G2P, as illustrated in Figure 2. SignTrak first decomposes the sequence-level Flow Matching objective into occurrence-level losses for attribution, and then aggregates attribution across query occurrences of the same gloss to guide occurrence-level reweighting.

3.1 OCCURRENCE-LEVEL DATA ATTRIBUTION

Occurrence-level attribution requires measuring how an individual training occurrence affects a query occurrence of the same gloss. However, the standard Flow Matching objective supervises the complete gloss–pose sequence and does not provide an individual loss for each gloss occurrence. We therefore first decompose the sequence-level objective into occurrence-level losses, whose gradients can then be used for attribution.

Sensitivity-guided loss decomposition. Deriving occurrence level losses requires determining which temporal losses correspond to each gloss occurrence. However, G2P datasets do not provide frame-level gloss–pose alignment, making explicit temporal segmentation difficult. Moreover, transitions and co-articulation allow neighboring glosses to jointly affect the same motion, making a hard assignment of each latent position to a single occurrence overly restrictive. We use conditioning sensitivity to divide the temporal loss among occurrences. However, sensitivity may also capture long-range contextual dependence beyond temporally adjacent glosses, so we impose an ordered temporal constraint to preserve gloss order.

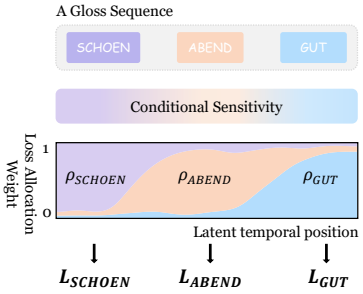


Figure 3: Sensitivity-guided loss decomposition.

Let $o_{n,k}$ denote the k-th gloss occurrence in G_n. To measure its influence on the generated pose sequence, we perturb its embedding along the direction from a null embedding to its original embedding using a scalar γ, while keeping the remaining gloss embeddings unchanged. Specifically, $e_{n,k}(\gamma) = e_{\text{null},k} + \gamma(e_{n,k} - e_{\text{null},k})$, where $\gamma = 1$ recovers the original embedding. We measure sensitivity by how the predicted velocity changes with respect to this perturbation. For brevity, $v_\theta(x_t, t, G_n; \gamma)$ denotes the velocity prediction under this perturbation of occurrence k. For the r-th sampled Flow Matching state, we define

$$s^{(r)}_{n,k,m} = \left\| \frac{\partial [v_\theta(x_t^{(r)}, t^{(r)}, G_n; \gamma)]_m}{\partial \gamma} \Big|_{\gamma=1} \right\|_2^2. \tag{5}$$

Here, $\gamma = 1$ corresponds to the original gloss embedding, and a larger $s^{(r)}_{n,k,m}$ indicates greater sensitivity of position m to occurrence k. To preserve temporal order, we use a monotonic path that assigns each latent position to either one gloss occurrence or two adjacent occurrences. The

path is selected by Viterbi decoding to maximize the accumulated sensitivity score under the gloss-order constraint. Let $A_{n,m}$ denote the occurrence(s) allowed at position m. For each sampled Flow Matching state, we normalize the sensitivities among the occurrences in $A_{n,m}$ and then average across states. When the selected sensitivities are too small for reliable allocation, the corresponding loss is left unassigned. For assigned states, the allocation is

$$\rho_{n,k,m} = \frac{1}{R}\sum_{r=1}^{R}\begin{cases}\dfrac{s^{(r)}_{n,k,m}}{\sum_{j\in A_{n,m}} s^{(r)}_{n,j,m}}, & k \in A_{n,m},\\ 0, & \text{otherwise},\end{cases} \tag{6}$$

where an unassigned state contributes zero to all occurrences. Thus, $\rho_{n,k,m}$ specifies the fraction of the temporal loss at position m assigned to occurrence k. We denote the unassigned fraction by $\rho_{n,0,m} = 1 - \sum_{k=1}^{K_n} \rho_{n,k,m}$. Using the temporal Flow Matching loss ℓ_m in Eq. 3, the occurrence-level loss is

$$\mathcal{L}_{\text{occ}}(o_{n,k};\theta) = \mathbb{E}_{t,\epsilon}\left[\sum_{m=1}^{L_n} \text{sg}(\rho_{n,k,m})\ell_m(z_n;\theta,H,t,\epsilon)\right]. \tag{7}$$

Here, $\text{sg}(\cdot)$ denotes stop-gradient. The allocation ratios are computed once using the pretrained generator and kept fixed during attribution and fine-tuning. Together with the unassigned fraction, the allocation preserves the original sequence-level Flow Matching loss.

Gradient-based influence estimation. Given occurrence-level losses, we estimate attribution through their gradients. If a training occurrence and a query occurrence have aligned loss gradients, updating the model using the training occurrence is expected to reduce the query loss. Let o and q denote a training and a query occurrence of the same gloss, respectively. Only the occurrence-assigned loss is used for attribution; the unassigned part is not attributed to any occurrence.

Following TRAK (Park et al., 2023), we use random projection to reduce the dimensionality of occurrence-loss gradients. For either a training or query occurrence $a \in \{o, q\}$, we construct

$$\phi(a) = \frac{1}{S}\sum_{s=1}^{S}\frac{P^\top \nabla_\psi \mathcal{L}^{(s)}_{\text{occ}}(a;\theta)}{b_s}. \tag{8}$$

Here, ψ denotes the selected velocity-head parameters, P is a shared random projection, and b_s accounts for scale differences across sampled Flow Matching states. The same projection and scaling are used for training and query occurrences. We define the occurrence-level attribution score as

$$\widehat{\mathcal{I}}(o,q) = \phi(q)^\top \phi(o). \tag{9}$$

A positive score indicates aligned gradients and a beneficial influence on the query occurrence, while a negative score indicates the opposite effect. These scores are subsequently used to compare and reweight training occurrences of the same gloss.

3.2 Attribution-Guided Optimization

The attribution score in Section 3.1 measures how a training occurrence affects a particular query occurrence of the same gloss. To guide optimization, we need to convert these pairwise scores into a training weight for each occurrence. However, the same training occurrence may have different effects on different query occurrences, and independently reweighting its occurrence-level loss may change the overall contribution of that gloss to the training objective. We therefore first aggregate attribution over multiple query occurrences of the same gloss, and then jointly reweight its training occurrences so that weight increases for some occurrences are balanced by decreases for others.

Attribution aggregation. We average the attribution of a training occurrence over multiple query occurrences of the same gloss to measure its overall usefulness. For a gloss g, let $\mathcal{Q}_g$ denote its query occurrences. We define the utility of training occurrence o as

$$U(o) = \frac{1}{|\mathcal{Q}_g|}\sum_{q\in\mathcal{Q}_g}\widehat{\mathcal{I}}(o,q). \tag{10}$$

Since positive attribution indicates a beneficial influence, a larger $U(o)$ means that occurrence o is, on average, more useful across query occurrences of the same gloss.

Occurrence reweighting. We next convert the aggregated utilities into training weights, assigning larger weights to more useful occurrences and smaller weights to less useful ones. Different occurrences receive different fractions of the original sequence-level loss. For occurrence $o_{n,k}$, we summarize its assigned fraction by $\mu(o_{n,k}) = L_n^{-1} \sum_{m=1}^{L_n} \rho_{n,k,m}$. Let $\mathcal{O}_g$ denote the training occurrences of gloss g with $\mu(o) > 0$. Within each gloss, we normalize the utilities and clip extreme values before converting them into candidate weights c_o. A larger utility produces a larger candidate weight, while $c_o = 1$ corresponds to the original training weight. We denote the final training weight for occurrence o by α_o. Starting from the candidate weights c_o, we jointly adjust the weights of occurrences belonging to the same gloss:

$$\min_{\{\alpha_o\}} \frac{1}{2} \sum_{o \in \mathcal{O}_g} \mu(o)(\alpha_o - c_o)^2 \quad \text{s.t.} \quad \sum_{o \in \mathcal{O}_g} \mu(o)(\alpha_o - 1) = 0. \tag{11}$$

The objective keeps the final weights close to the candidate weights, while the constraint balances weight increases and decreases among occurrences of the same gloss. We additionally restrict $\alpha_o \in [\alpha_{\min}, \alpha_{\max}]$ to avoid excessively large weight changes.

Let $\mathcal{L}_{\text{un}}(z_n; \theta)$ denote the unassigned part of the sequence-level loss induced by $\rho_{n,0,m}$; its definition is given in Appendix C.3. We fine-tune the generator using

$$\mathcal{L}_{\text{final}}(\theta) = \frac{1}{N} \sum_{n=1}^{N} \left[\mathcal{L}_{\text{un}}(z_n; \theta) + \sum_{k=1}^{K_n} \alpha_{o_{n,k}} \mathcal{L}_{\text{occ}}(o_{n,k}; \theta) \right]. \tag{12}$$

The unassigned loss always keeps unit weight. Occurrences with $\mu(o) = 0$, as well as glosses with too few training or query occurrences for reweighting, keep their original weight of one. When all occurrence weights satisfy $\alpha_{o_{n,k}} = 1$, we have $w_{n,m} = 1$, recovering the original Flow Matching objective. During fine-tuning, the model still takes the complete gloss–pose sequence as input; only the occurrence-assigned loss is reweighted.

4 EXPERIMENTS

We evaluate SignTrak from two complementary perspectives. First, we examine whether occurrence-level attribution can be converted into better G2P training through attribution-guided reweighting. Second, we evaluate whether the estimated attribution scores faithfully identify the contribution of individual training occurrences to a target gloss occurrence. Unless otherwise specified, attribution experiments, ablations, and qualitative analyses are conducted on Phoenix14T.

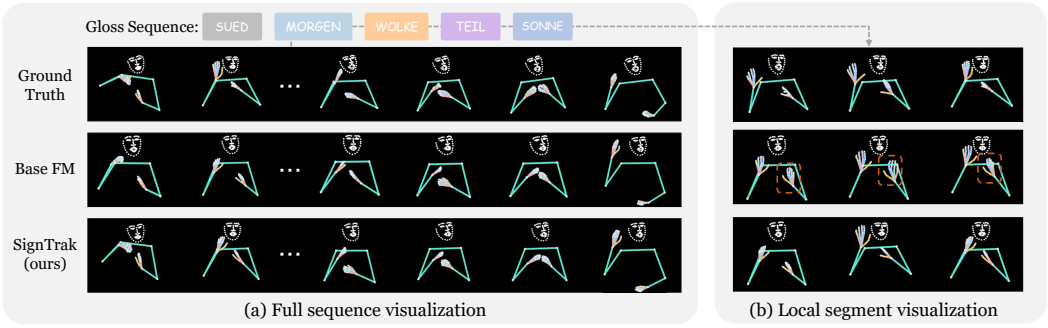
Datasets and Metrics. We evaluate SignTrak on Phoenix14T (Camgoz et al., 2018) and CSL-Daily (Zhou et al., 2021). Phoenix14T contains 8,257 German Sign Language videos (7,096/519/642 train/val/test) with 1,066 glosses, while CSL-Daily contains 20,654 Chinese Sign Language videos (18,401/1,077/1,176 train/val/test) with 2,000 glosses. We represent frames using HRNet-estimated 2D keypoints (Sun et al., 2019): 42 hand, 68 facial, and 11 upper-body keypoints.

For G2P evaluation, we use a fixed MixSignGraph (Gan et al., 2025) for back-translation and report WER, ROUGE-L, BLEU-1 and BLEU-4 for semantic fidelity, together with Mean Per Joint Position Error (MPJPE) for pose accuracy. For attribution evaluation, Spearman's ρ and Kendall's τ_b measure ranking agreement between predicted and empirical influence, while Precision@10 evaluates top-ranked attribution accuracy. For controlled corruption, we report Precision@10, Recall@10, and AUROC to measure the detection of corrupted training occurrences.

Implementation Details. (1) *Flow Matching-based G2P Model*: Our generator uses a 16-layer Transformer with a hidden dimension of 1,024, 16 attention heads, and an FFN expansion ratio of 4. The gloss encoder consists of 4 Transformer layers and a 6-layer adapter, while the frozen motion VAE uses 16-dimensional latents with a temporal downsampling factor of 4. We train the model for 500 epochs using AdamW with a peak learning rate of 1×10^{-4} and cosine decay, and use the Heun solver with 50 steps and a classifier-free guidance scale of 4.0 at inference. (2) *Occurrence Attribution*: We use 4 flow states for loss allocation ratio estimation and 8 shared flow states for influence estimation. Gradients are extracted from the final velocity-head layer and projected to 2,048 dimensions using SRHT. (3) *Attribution-Guided Optimization*: We constrain occurrence weights to

Table 1: G2P generation performance on Phoenix14T and CSL-Daily. † denotes reproduced results.

	Phoenix14T					CSL-Daily				
Method	WER↓	ROUGE↑	BLEU-1↑	BLEU-4↑	MPJPE↓	WER↓	ROUGE↑	BLEU-1↑	BLEU-4↑	MPJPE↓
PT† (Saunders et al., 2020b)	98.78	15.62	16.57	1.80	34.53	99.07	10.10	11.81	0.17	27.38
PT-GN† (Saunders et al., 2020b)	98.78	15.65	16.62	1.88	34.53	99.07	10.11	11.75	0.17	27.29
G2P-DDM† (Xie et al., 2024)	95.63	20.37	19.40	6.25	–	98.25	12.01	13.55	2.20	–
FGDM† (Yu et al., 2026)	92.49	22.71	22.30	5.41	32.03	79.04	23.93	25.82	5.57	22.84
Sign-IDD† (Tang et al., 2025a)	84.87	26.95	26.84	7.46	21.57	73.72	25.23	29.39	6.88	**19.17**
Base FM	58.40	40.35	39.57	15.12	22.81	55.49	36.49	38.50	12.15	21.77
SignTrak	**45.67**	**42.98**	**42.61**	**17.61**	**21.30**	**44.73**	**40.82**	**42.17**	**15.47**	21.08



(a) Full sequence visualization (b) Local segment visualization

Figure 4: Qualitative comparison of Base FM and SignTrak on Phoenix14T. We show both the complete generated sequence and a highlighted local pose sequence.

$[0.5, 1.5]$ and fine-tune the generator for 30 epochs with a peak learning rate of 1×10^{-5}. The batch size is 4 per GPU, and all experiments are conducted on four NVIDIA RTX A6000 GPUs.

4.1 G2P Generation Performance

Comparison. We first evaluate whether occurrence-level attribution can improve G2P generation. We train the Base FM model and then fine-tune it using attribution-guided optimization to obtain SignTrak. As shown in Table 1, SignTrak improves the pretrained Flow Matching baseline and achieves the best back-translation results among the compared methods. On Phoenix14T, SignTrak reduces WER (-12.73) and MPJPE (-1.51), improving both semantic and pose accuracy. On CSL-Daily, SignTrak also outperforms the reproduced prior methods on most metrics.

Global and Local Visualization. As shown in Figure 4, the left side presents selected key frames from different stages of the generated sequence, where both Base FM and SignTrak capture the overall motion trend of the ground truth. The right side further visualizes a consecutive local segment extracted from the omitted region in the left side. Within this segment, Base FM exhibits pose deviations across several consecutive frames, while these errors are reduced after attribution-guided fine-tuning with SignTrak. Additional visualization results are provided in Appendix E.

4.2 Occurrence-Level Attribution

We next evaluate whether SignTrak correctly ranks training occurrences according to their effect on a query occurrence. Since real G2P datasets do not provide ground-truth influence, we use two complementary evaluations: intervention predictivity and controlled occurrence corruption.

Controlled Occurrence Corruption. We randomly corrupt 5% of training occurrences by replacing their gloss labels while keeping the paired poses unchanged, with at most one corruption per sentence. Starting from the same clean FM pretrained model, we fine-tune on each corrupted training set for five epochs. We rank training occurrences from least to most useful according to their aggregated attribution scores and evaluate whether the corrupted occurrences are ranked near the least useful end using P@10, R@10, and AUROC. We compare against sequence-level gradient similarity (*Seq. GradSim*) and sequence-level TRAK (*Seq. TRAK*). As shown in Table 2, SignTrak identifies corrupted occurrences more reliably than sequence-level attribution methods, improving P@10 from 14.32 to 20.20 and achieving an AUROC of 60.34. Since only one occurrence in a

Table 2: Detection of corrupted training occurrences.

Method	P@10↑	R@10↑	AUROC↑
Random	13.49	9.85	50.00
Seq. GradSim	14.32	10.40	47.14
Seq. TRAK	11.65	8.50	46.87
SignTrak	**20.20**	**14.80**	**60.34**

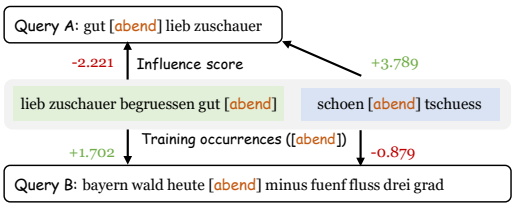


Figure 5: Case study of occurrence-level attribution.

sentence is corrupted while the remaining occurrences are unchanged, sequence-level scores can be dominated by the valid parts of the sequence. In contrast, SignTrak assigns attribution directly to occurrences, allowing corrupted occurrences to be ranked toward the low-score end.

Case Study of Occurrence-Level Attribution. Figure 5 shows that the influence of a training occurrence can vary across query occurrences of the same gloss. In this case, the *[abend]* occurrence in *lieb zuschauer begruessen gut [abend]* has a negative influence on Query A (-2.221) but a positive influence on Query B ($+1.702$), while the other training occurrence exhibits the opposite trend. This influence reversal shows that the usefulness of a training occurrence cannot be determined from gloss identity alone, motivating aggregation over multiple query occurrences when computing $U(o)$.

Intervention Predictivity. For each query occurrence, we consider candidate training occurrences of the same gloss. Starting from the same pretrained FM checkpoint, we slightly upweight each candidate, perform a short optimization, and measure the resulting reduction in query occurrence loss. We then compare this ranking with the ranking given by each attribution method using Spearman's ρ, Kendall's τ_b, and P@10. For the sequence-level baselines, each candidate uses the attribution score of its training sequence, whereas SignTrak directly computes the score between the training occurrence and the query occurrence. As shown in Table 3, SignTrak outperforms sequence-level attribution methods across all three metrics, with Spearman's ρ increasing from 0.0500 to 0.241 over the strongest baseline. Sequence-level attribution assigns each candidate the score of its containing sequence, which also reflects other glosses in that sequence. In contrast, SignTrak directly compares occurrence-level losses, allowing its ranking to better match the observed effect of upweighting individual training occurrences.

Table 3: Attribution predictivity.

Method	ρ↑	τ_b↑	P@10↑
Seq. GradSim	0.0500	0.0364	0.2433
Seq. TRAK	0.0279	0.0216	0.2250
SignTrak	**0.241**	**0.1728**	**0.3329**

4.3 Ablation and Analysis

Table 4: Ablation study.

Variant	Attr. ρ↑	WER↓	MPJPE↓
Sample-level Attribution	0.098	46.34	**21.26**
CTC Loss Decomposition	0.064	47.10	22.21
Uniform FT	–	56.65	22.74
Single Query	0.036	46.26	22.22
SignTrak	**0.241**	**45.67**	21.30

Component Ablation. We conduct ablations on the two main components of SignTrak. For occurrence-level data attribution, *Sample-level Attribution* uses the complete training sequence as the attribution unit, while *CTC-based Assignment* replaces our sensitivity-guided decomposition with CTC-based temporal assignment (Graves et al., 2006). For attribution-guided optimization, *Single Query* uses only one query occurrence, while *Uniform FT* continues training Base FM without attribution-guided reweighting. As shown in Table 4, sample-level attribution cannot distinguish individual gloss occurrences within the same sequence, while Single Query is sensitive to the query context. CTC-based assignment also performs worse than our decomposition, indicating the benefit of sharing loss around transitions. Uniform FT performs much worse, showing that the gains come from attribution-guided reweighting rather than additional fine-tuning. Overall, SignTrak achieves the best attribution quality and WER, while remaining comparable in MPJPE.

Analysis of Sensitivity-Guided Loss Decomposition. We further compare our sensitivity-guided loss decomposition with CTC-based temporal assignment. As shown in Figure 6, both methods

produce temporally ordered assignments that broadly follow the gloss sequence. CTC assigns each temporal position to a single gloss occurrence, leading to relatively sharp changes around estimated boundaries. In comparison, our method allows the loss at transition regions to be shared between two adjacent occurrences according to their conditioning sensitivities, while positions mainly associated with one gloss remain concentrated on that occurrence. This yields occurrence-level losses that remain temporally localized without forcing transition regions to a single gloss occurrence.

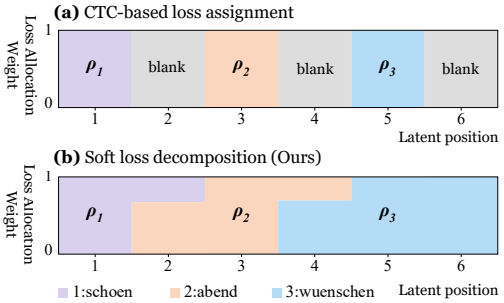


Figure 6: CTC-based temporal assignment versus sensitivity-guided loss decomposition.

5 Related work

Sign Language Production. Sign Language Production (SLP) (Saunders et al., 2020a; 2022; Stoll et al., 2022; Fang et al., 2023; Low et al., 2026) aims to generate continuous sign language pose sequences from spoken-language text. Existing methods can be divided into two pipelines: *text2pose* and *text2gloss2pose*. (1) Text2pose methods directly learn the mapping from text to pose sequences or latent pose representations using generative models such as Transformers and diffusion models (Zuo et al., 2024; Baltatzis et al., 2024; Yin et al., 2024). However, text and sign language pose are not monotonically aligned in time, increasing the difficulty of learning this mapping. (2) Text2gloss2pose methods introduce glosses as an intermediate representation. Since gloss order generally follows the temporal order of signing motion, glosses provide a more direct correspondence with pose sequences. Gloss2Pose (G2P) methods are mainly dictionary-based or generative. Dictionary-based methods retrieve and concatenate gloss-associated motion segments (Zuo et al., 2025; Walsh et al., 2024b), whereas generative methods learn the gloss-to-pose mapping (Xie et al., 2024; Yu et al., 2026; Tang et al., 2025a). However, existing G2P methods mainly rely on sequence-level supervision, making it difficult to improve poorly generated pose segments corresponding to particular glosses. We address this problem through occurrence-level data attribution.

Data Attribution Data attribution (Wang et al., 2024; Murata et al., 2026; Mlodozeniec et al., 2026) studies how training samples affect a model's prediction on a query. Classical approaches estimate this influence through sample removal or reweighting (Koh & Liang, 2017; Pruthi et al., 2020), while scalable methods such as TRAK (Park et al., 2023) approximate it with projected gradients. Recent work extends data attribution to diffusion models, where influence estimation must account for the stochastic generation process and variation across denoising steps (Zheng et al., 2024). Beyond whole-sample attribution, recent studies have considered finer attribution units, including semantic concepts (Park et al., 2025), motion (Wu et al., 2026) and spatial regions (Nguyen et al., 2025). Despite considering more specific generated content, existing methods still estimate influence at the level of complete training samples. In G2P, however, each training sample contains multiple gloss occurrences, and a single sample-level score cannot isolate the influence of a particular occurrence. We therefore study occurrence-level data attribution for G2P.

6 Conclusion

In this work, we study G2P generation from the perspective of training data and introduce SignTrak, an occurrence-level data attribution and optimization framework. To investigate how individual gloss occurrences within training sequences affect the generation of a particular gloss, we introduce occurrence-level data attribution. SignTrak derives occurrence-level losses from sequence-level supervision using gradient sensitivity, which enables attribution to individual training occurrences. To further use this influence for G2P optimization, SignTrak aggregates attribution across query occurrences of the same gloss and reweights individual training occurrences during fine-tuning. Experiments show that SignTrak provides more accurate occurrence-level attribution than sequence-level baselines and improves G2P generation performance.

AI USE STATEMENT

In this work, we did not use generative AI tools for tasks requiring disclosure, and the remaining required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for language editing, including improving grammar, readability, and clarity, identifying writing errors, and refining the presentation and layout of figures and tables. We reviewed all AI-assisted edits to ensure that they preserve the intended technical meaning and do not introduce unsupported claims. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work uses existing publicly available sign language datasets, Phoenix14T and CSL-Daily, and does not collect new sign language recordings. We follow the usage conditions of the source datasets. This work also includes a user study in which participants evaluate generated sign language pose sequences. Participants were informed of the study procedure before participation, and no personally identifying information was collected. The responses were used only for evaluating the generated pose sequences. As the experiments rely on existing datasets, the resulting models may inherit limitations in signer coverage and data distribution from these datasets.

REPRODUCIBILITY STATEMENT

We provide the model architecture, training configuration, data preprocessing, evaluation protocols, and main hyperparameters in the main paper. Additional implementation details are provided in the Appendix, including the settings for sensitivity-guided loss decomposition, projected occurrence attribution, and attribution-guided reweighting. The Appendix also describes the additional evaluation protocols used for the user study and local pose evaluation. Together, these details provide the information needed to reproduce the proposed method and the reported experiments.

REFERENCES

Vasileios Baltatzis, Rolandos Alexandros Potamias, Evangelos Ververas, Guanxiong Sun, Jiankang Deng, and Stefanos Zafeiriou. Neural sign actors: A diffusion model for 3d sign language production from text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1985–1995, 2024.

Léore Bensabath, Mathis Petrovich, and Gül Varol. Text-driven 3d hand motion generation from sign language data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23095–23105, 2026.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. Neural sign language translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7784–7793, 2018.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. Sign language transformers: Joint end-to-end sign language recognition and translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10023–10033, 2020.

Yutong Chen, Ronglai Zuo, Fangyun Wei, Yu Wu, Shujie Liu, and Brian Mak. Two-stream network for sign language recognition and translation. *Advances in Neural Information Processing Systems*, 35:17043–17056, 2022.

Sen Fang, Chunyu Sui, Xuedong Zhang, and Yapeng Tian. Signdiff: Learning diffusion models for american sign language production. *arXiv e-prints*, pp. arXiv–2308, 2023.

Fernando A Fardo, Victor H Conforto, Francisco C De Oliveira, and Paulo S Rodrigues. A formal evaluation of psnr as quality measurement parameter for image segmentation algorithms. *arXiv preprint arXiv:1605.07116*, 2016.

Shiwei Gan, Yafeng Yin, Zhiwei Jiang, Lei Xie, Sanglu Lu, and Hongkai Wen. Mixsigngraph: A sign sequence is worth mixed graphs of nodes. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Jia Gong, Lin Geng Foo, Yixuan He, Hossein Rahmani, and Jun Liu. Llms are good sign language translators. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18362–18372, 2024.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pp. 369–376, 2006.

Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10696–10706, 2022.

Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Li Hu. Animate anyone: Consistent and controllable image-to-video synthesis for character animation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8153–8163, 2024.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International conference on machine learning*, pp. 1885–1894. Pmlr, 2017.

Chin-Yew Lin and Franz Josef Och. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd annual meeting of the association for computational linguistics (ACL-04)*, pp. 605–612, 2004.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Xiao Liu, Shiwei Gan, Yafeng Yin, Bowen Guo, Zhiwei Jiang, Shunmei Meng, Lei Xie, and Sanglu Lu. Signpr: A progressive vector-quantized diffusion framework for sign language production. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2198–2208, 2026.

Jianhe Low, Alexandre Symeonidis-Herzig, Maksym Ivashechkin, Ozge Mercanoglu Sincan, and Richard Bowden. Signspark: Efficient multilingual sign language production via sparse keyframe learning. In *European Conference on Computer Vision*, pp. 648–670. Springer, 2026.

Jian Ma, Wenguan Wang, Yi Yang, and Feng Zheng. Ms2sl: multimodal spoken data-driven continuous sign language production. *arXiv preprint arXiv:2407.12842*, 2024.

Bruno Mlodozeniec, Isaac Reid, Sam Power, David Krueger, Murat Erdogdu, Richard Turner, and Roger Grosse. Distributional training data attribution: What do influence functions sample? *Advances in Neural Information Processing Systems*, 38:10057–10091, 2026.

Naoki Murata, Yuhta Takida, Chieh-Hsin Lai, Toshimitsu Uesaka, Bac Nguyen, Stefano Ermon, and Yuki Mitsufuji. Guda: Counterfactual group-wise training data attribution for diffusion models via unlearning. *arXiv preprint arXiv:2601.22651*, 2026.

Trong Bang Nguyen, Phi Le Nguyen, Simon Lucey, and Minh Hoai. Region-level data attribution for text-to-image generative models. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18825–18833. IEEE, 2025.

Sung Min Park, Kristian Georgiev, Andrew Ilyas, Guillaume Leclerc, and Aleksander Madry. Trak: Attributing model behavior at scale. *arXiv preprint arXiv:2303.14186*, 2023.

Yonghyun Park, Chieh-Hsin Lai, Satoshi Hayakawa, Yuhta Takida, Naoki Murata, Wei-Hsiang Liao, Woosung Choi, Kin Wai Cheuk, Junghyun Koo, and Yuki Mitsufuji. Concept-trak: Understanding how diffusion models learn concepts through concept-level attribution. *arXiv preprint arXiv:2507.06547*, 2025.

Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating training data influence by tracing gradient descent. *Advances in neural information processing systems*, 33: 19920–19930, 2020.

Fan Qi, Yu Duan, Huaiwen Zhang, and Changsheng Xu. Signgen: End-to-end sign language video generation with latent diffusion. In *European Conference on Computer Vision*, pp. 252–270. Springer, 2024.

Razieh Rastgoo, Kourosh Kiani, Sergio Escalera, and Mohammad Sabokrou. Sign language production: A review. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3451–3461, 2021.

Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

Ben Saunders, Necati Cihan Camgoz, and Richard Bowden. Adversarial training for multi-channel sign language production. *arXiv preprint arXiv:2008.12405*, 2020a.

Ben Saunders, Necati Cihan Camgoz, and Richard Bowden. Progressive transformers for end-to-end sign language production. In *European Conference on Computer Vision*, pp. 687–705. Springer, 2020b.

Ben Saunders, Necati Cihan Camgoz, and Richard Bowden. Continuous 3d multi-channel sign language production via progressive transformers and mixture density networks. *International journal of computer vision*, 129(7):2113–2135, 2021a.

Ben Saunders, Necati Cihan Camgoz, and Richard Bowden. Mixed signals: Sign language production via a mixture of motion primitives. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1919–1929, 2021b.

Ben Saunders, Necati Cihan Camgoz, and Richard Bowden. Signing at scale: Learning to co-articulate signs for large-scale photo-realistic sign language production. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5141–5151, 2022.

Bowen Shi, Diane Brentari, Gregory Shakhnarovich, and Karen Livescu. Open-domain sign language translation learned from online video. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 6365–6379, 2022.

Tongkai Shi, Lianyu Hu, Fanhua Shang, Jichao Feng, Peidong Liu, and Wei Feng. Pose-guided fine-grained sign language video generation. In *European Conference on Computer Vision*, pp. 392–409. Springer, 2024.

Stephanie Stoll, Necati Cihan Camgöz, Simon Hadfield, and Richard Bowden. Sign language production using neural machine translation and generative adversarial networks. In *Proceedings of the 29th British Machine Vision Conference (BMVC 2018)*. British Machine Vision Association, 2018.

Stephanie Stoll, Armin Mustafa, and Jean-Yves Guillemaut. There and back again: 3d sign language generation from text using back-translation. In *2022 International Conference on 3D Vision (3DV)*, pp. 187–196. IEEE, 2022.

Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, pp. 5693–5703, 2019.

Shengeng Tang, Jiayi He, Dan Guo, Yanyan Wei, Feng Li, and Richang Hong. Sign-idd: Iconicity disentangled diffusion for sign language production. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 7266–7274, 2025a.

Shengeng Tang, Feng Xue, Jingjing Wu, Shuo Wang, and Richang Hong. Gloss-driven conditional diffusion models for sign language production. *ACM Transactions on Multimedia Computing, Communications and Applications*, 21(4):1–17, 2025b.

Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018.

Ruben Villegas, Mohammad Babaeizadeh, Pieter-Jan Kindermans, Hernan Moraldo, Han Zhang, Mohammad Taghi Saffar, Santiago Castro, Julius Kunze, and Dumitru Erhan. Phenaki: Variable length video generation from open domain textual description. *arXiv preprint arXiv:2210.02399*, 2022.

Harry Walsh, Abolfazl Ravanshad, Mariam Rahmani, and Richard Bowden. A data-driven representation for sign language production. In *2024 IEEE 18th International Conference on Automatic Face and Gesture Recognition (FG)*, pp. 1–10. IEEE, 2024a.

Harry Walsh, Ben Saunders, and Richard Bowden. Sign stitching: A novel approach to sign language production. *arXiv preprint arXiv:2405.07663*, 2024b.

Cong Wang, Zexuan Deng, Zhiwei Jiang, Yafeng Yin, Fei Shen, Zifeng Cheng, Shiping Ge, Shiwei Gan, and Qing Gu. Advanced sign language video generation with compressed and quantized multi-condition tokenization. *arXiv preprint arXiv:2506.15980*, 2025a.

Sheng-Yu Wang, Aaron Hertzmann, Alexei A Efros, Jun-Yan Zhu, and Richard Zhang. Data attribution for text-to-image models by unlearning synthesized images. *Advances in Neural Information Processing Systems*, 37:4235–4266, 2024.

Xu Wang, Shengeng Tang, Peipei Song, Shuo Wang, Dan Guo, and Richang Hong. Linguistics-vision monotonic consistent network for sign language production. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2025b.

Fangyun Wei and Yutong Chen. Improving continuous sign language recognition with cross-lingual signs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 23612–23621, 2023.

Xindi Wu, Despoina Paschalidou, Jun Gao, Antonio Torralba, Laura Leal-Taixé, Olga Russakovsky, Sanja Fidler, and Jonathan Lorraine. Motion attribution for video generation. *arXiv preprint arXiv:2601.08828*, 2026.

Pan Xie, Qipeng Zhang, Peng Taiying, Hao Tang, Yao Du, and Zexian Li. G2p-ddm: Generating sign pose sequence from gloss sequence with discrete diffusion model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 6234–6242, 2024.

Aoxiong Yin, Tianyun Zhong, Li Tang, Weike Jin, Tao Jin, and Zhou Zhao. Gloss attention for gloss-free sign language translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2551–2562, 2023.

Aoxiong Yin, Haoyuan Li, Kai Shen, Siliang Tang, and Yueting Zhuang. T2s-gpt: Dynamic vector quantization for autoregressive sign language production from text. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3345–3356, 2024.

Yiheng Yu, Sheng Liu, Yuan Feng, Zhelun Jin, Yining Jiang, and Min Xu. Focal-general diffusion model with semantic consistent guidance for sign language production. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 35915–35925, 2026.

Xiaosen Zheng, Tianyu Pang, Chao Du, Jing Jiang, and Min Lin. Intriguing properties of data attribution on diffusion models. In *International Conference on Learning Representations*, volume 2024, pp. 18417–18452, 2024.

Hao Zhou, Wengang Zhou, Weizhen Qi, Junfu Pu, and Houqiang Li. Improving sign language translation with monolingual data by sign back-translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1316–1325, 2021.

Ronglai Zuo, Fangyun Wei, Zenggui Chen, Brian Mak, Jiaolong Yang, and Xin Tong. A simple baseline for spoken language to sign language translation with 3d avatars. In *European Conference on Computer Vision*, pp. 36–54. Springer, 2024.

Ronglai Zuo, Rolandos Alexandros Potamias, Evangelos Ververas, Jiankang Deng, and Stefanos Zafeiriou. Signs as tokens: A retrieval-enhanced multilingual sign language generator. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 23806–23816, 2025.

A Discussion and Limitations

Discussion. Our experiments reveal several findings about training occurrences in G2P. (1) Our attribution examples show that the same training occurrence can help one query but hurt another occurrence of the same gloss. This indicates that its usefulness depends on the surrounding context, rather than on gloss identity alone. (2) When occurrence-level attribution is weakened or replaced by coarser alternatives, the final generation performance also becomes worse. This suggests that distinguishing useful training occurrences is important for assigning effective training weights. (3) We observe a larger improvement in WER than in average pose error. A likely reason is that correcting a short local motion segment has only a small effect on an error averaged over the whole sequence, showing the importance of evaluating local pose quality.

Limitations. Our current method still has several limitations. (1) SignTrak requires enough training and query occurrences for each gloss. For rare glosses, only a few occurrences are available for estimating their usefulness. (2) Attribution scores are computed before fine-tuning and remain fixed afterward. As the model changes during fine-tuning, the usefulness of individual training occurrences may also change. Updating the scores during training may improve this, but would require additional computation. (3) Our loss decomposition does not provide an exact temporal alignment between glosses and pose segments. The allocation is estimated from the pretrained model's conditioning sensitivity, so it may differ from the actual temporal region of a gloss.

B Additional Implementation Details

(1) Flow Matching-based G2P Model. Our generator uses a 16-layer Transformer with a hidden dimension of 1,024, 16 attention heads, an FFN expansion ratio of 4, and a dropout rate of 0.1. The gloss encoder consists of 4 Transformer layers followed by a 6-layer adapter, both using 8 attention heads and a dropout rate of 0.1. The pretrained motion VAE uses 16-dimensional latents with a temporal downsampling factor of 4 and remains frozen throughout training and fine-tuning. The maximum pose and gloss sequence lengths are 512 frames and 64 tokens, respectively. We train the model for 500 epochs using AdamW with $(\beta_1, \beta_2) = (0.9, 0.95)$ and zero weight decay. The learning rate is linearly warmed up to 1×10^{-4} over the first 10 epochs and then decayed to 1×10^{-6} using a cosine schedule. At inference, we use the Heun solver with 50 integration steps and classifier-free guidance with a maximum scale of 4.0.

(2) Occurrence Attribution. We use 4 flow states to estimate the loss allocation ratios. For influence estimation, we use 8 shared flow states constructed from 4 quantile midpoints of the training timestep distribution and 2 Gaussian noise draws per timestep. Gradients are computed with respect to the parameters of the final velocity-head affine layer and projected to 2,048 dimensions using a subsampled randomized Hadamard transform (SRHT). For each flow state, projected gradients are normalized by the median projected-gradient norm estimated over occurrences from 512 training sequences before averaging across states. The same flow-state bank, projection matrix, and normalization scales are used for training and query occurrences. We use the official development split as the query set for attribution.

(3) Attribution-Guided Optimization. Occurrence utilities are standardized within each gloss using the median and median absolute deviation (MAD) and clipped to $[-3, 3]$. The reweighting strength is set to 0.25, and occurrence weights are constrained to $[0.5, 1.5]$. Glosses with fewer than 3 positive-mass training occurrences or no query occurrences retain unit weights. We fine-tune the generator for 30 epochs using AdamW with the same optimizer coefficients and weight decay as in base training. The learning rate is linearly warmed up to 1×10^{-5} over 2 epochs and then decayed to 1×10^{-6} using a cosine schedule. The global batch size is 16.

C Additional Method Details

This section provides additional details for the components introduced in Section 3. We describe how the ordered loss allocation is obtained, how gradients from different Flow Matching states are scaled for attribution, and how the resulting occurrence weights are applied during fine-tuning.

C.1 Sensitivity-Guided Loss Decomposition

To compute the conditioning sensitivity in Eq. (5), we perturb only the content of one gloss occurrence while keeping its positional encoding unchanged. Specifically, the input embedding is $e_{n,k}(\gamma) = e_{\mathrm{null},k} + \gamma(e_{n,k} - e_{\mathrm{null},k})$, where $e_{\mathrm{null},k}$ uses the same positional encoding as $e_{n,k}$. Thus, $\gamma = 1$ recovers the original input. To construct the allowed occurrence set $A_{n,m}$ in Eq. (6), we first normalize the conditioning sensitivities across gloss occurrences for each sampled Flow Matching state and aggregate them across states. For a sequence with K gloss occurrences, the ordered path contains states $\{1\}, \{1,2\}, \{2\}, \ldots, \{K-1, K\}, \{K\}$. A state therefore contains either one occurrence or two adjacent occurrences. The path can stay at the current state or advance to the next state, and Viterbi decoding selects the path with the largest accumulated sensitivity score. The selected state at latent position m defines $A_{n,m}$. For shorter latent sequences, we use a reduced ordered path while preserving the same gloss order.

At some latent positions, the selected occurrences have very small conditioning sensitivity. We do not assign such loss to a particular occurrence. Instead, let $\rho_{n,0,m}$ denote the fraction that remains unassigned after averaging over Flow Matching states. The complete allocation satisfies

$$\rho_{n,0,m} + \sum_{k=1}^{K_n} \rho_{n,k,m} = 1. \tag{13}$$

The occurrence-assigned fractions $\rho_{n,k,m}$ are used in $\mathcal{L}_{\mathrm{occ}}$, while the unassigned fraction keeps weight one during fine-tuning.

C.2 Projected Occurrence Attribution

We further specify the state-wise scale b_s used in Eq. (8). For Flow Matching state s, $\mathcal{L}_{\mathrm{occ}}^{(s)}$ denotes Eq. (7) evaluated using the corresponding fixed (t_s, ϵ_s). We compute

$$b_s = \mathrm{median}_{o \in \mathcal{B}} \left\| P^\top \nabla_\psi \mathcal{L}_{\mathrm{occ}}^{(s)}(o; \theta) \right\|_2, \tag{14}$$

where $\mathcal{B}$ is a fixed subset of training occurrences. Each projected gradient is divided by b_s before the gradients are averaged across Flow Matching states. The same states, projection matrix, and scale factors are used for training and query occurrences.

C.3 Attribution-Guided Reweighting

Within each gloss, the utilities $U(o)$ are normalized using the median and median absolute deviation (MAD), clipped to a fixed range, and mapped linearly to the candidate weights c_o centered at one. If the utilities of a gloss have negligible variation, its candidate weights are kept at one.

For the constrained reweighting in Eq. (11), the final occurrence weights can be written as $\alpha_o = \mathrm{clip}(c_o - \lambda_g, \alpha_{\min}, \alpha_{\max})$, where the same offset λ_g is used for all occurrences of gloss g. We choose λ_g by one-dimensional bisection so that the constraint in Eq. (11) is satisfied. Thus, the weights remain within the specified range while $\sum_{o \in O_g} \mu(o)\alpha_o = \sum_{o \in O_g} \mu(o)$. The unassigned loss in Eq. 12 is defined as

$$\mathcal{L}_{\mathrm{un}}(z_n; \theta) = \mathbb{E}_{t,\epsilon}\left[\sum_{m=1}^{L_n} \mathrm{sg}(\rho_{n,0,m}) \ell_m(z_n; \theta, H, t, \epsilon) \right]. \tag{15}$$

During fine-tuning, the fixed allocation ratios and occurrence weights can be equivalently combined into a temporal loss weight:

$$w_{n,m} = \rho_{n,0,m} + \sum_{k=1}^{K_n} \alpha_{o_{n,k}} \rho_{n,k,m}. \tag{16}$$

The temporal Flow Matching loss ℓ_m is multiplied by $w_{n,m}$ before summing over latent positions. The unassigned part therefore keeps weight one, while the occurrence-assigned parts use their corresponding occurrence weights. When all $\alpha_o = 1$, we have $w_{n,m} = 1$, recovering the original Flow Matching objective.

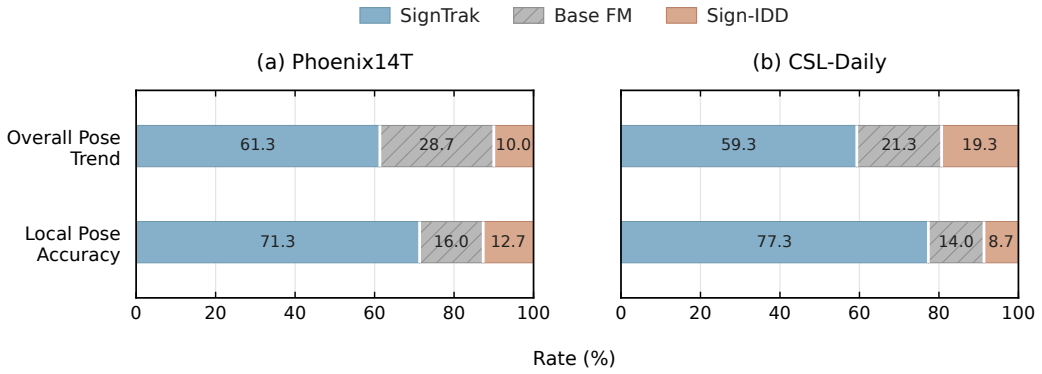


Figure 7: User study.

D Additional Analysis

User Study. We conduct a reference-based user study with 10 non-expert participants to compare the generated pose sequences. We randomly sample 15 test sequences from Phoenix14T and 15 from CSL-Daily. For each sample, participants are shown the ground-truth (GT) pose sequence together with three corresponding sequences generated by SignTrak, Base FM, and Sign-IDD. GT is labeled as the reference, while the three generated sequences are anonymized and shown in random order. Participants make two comparisons: (1) *Overall Pose Trend*, selecting the generated sequence whose overall motion is most similar to GT; and (2) *Local Pose Accuracy*, selecting the generated sequence whose poses are most similar to GT within a fixed local segment. The same local segment is highlighted for GT and all generated sequences. For each comparison, participants select one of the three generated sequences, with no tie option. We report the percentage of selections received by each method on each dataset.

Figure 7 shows that SignTrak is preferred more often than the two baselines on both datasets. The difference is larger for local pose accuracy than for overall pose trend, indicating that SignTrak provides a clearer improvement in the accuracy of local motion segments. This result is consistent with the goal of occurrence-level optimization, which aims to improve local pose quality within the generated sequence.

Local Gloss-Level Evaluation. To evaluate local motion quality, we use a frozen CTC-based pose-to-gloss recognizer to align the reference and generated sequences separately to the ground-truth gloss transcript. For each corresponding gloss occurrence, we temporally resample the generated segment to the reference length and compute MPJPE over body and hand joints. We evaluate all gloss occurrences in the Phoenix14T test set and report local MPJPE together with hand-only and body-only errors. As shown in Table 5, SignTrak reduces local MPJPE by 6.10% relative to Base FM. The reductions are similar for hand and body joints (6.17% and 6.42%), indicating that the improvement is not limited to a single joint group. Since the evaluated regions are obtained independently of SignTrak's loss allocation, these results provide direct evidence that the improvement extends to gloss-local pose generation.

Table 5: Local gloss-level pose errors.

Metric ↓	Base FM	SignTrak	Reduction
Local MPJPE	30.81	**28.93**	6.10%
Hand MPJPE	33.21	**31.16**	6.17%
Body MPJPE	16.82	**15.74**	6.42%

E Additional Qualitative Results

Gloss-level error analysis. We further analyze how pose errors are distributed within a generated sequence. As shown in Figure 8, the error often remains relatively low over most frames but rises sharply around a short CTC-aligned gloss region. Across the test set,

Table 6: Localized error analysis on Phoenix14T.

	Local	Rest	Ratio
Mean error	0.480	0.258	1.86×
Ratio $\geq$ 1.5: 85.33%		Ratio $\geq$ 2.0: 30.33%	

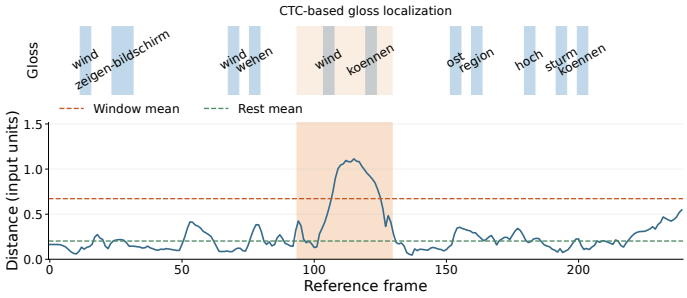


Figure 8: Visualization of Localized Errors.

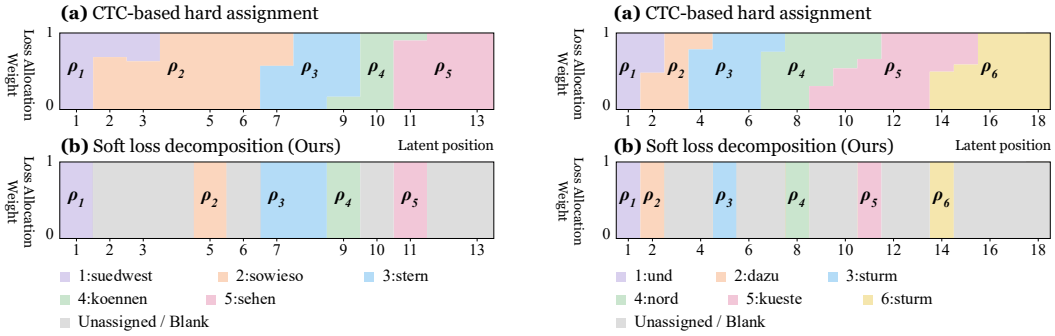


Figure 9: Sensitivity-Guided Loss Decomposition examples.

the selected local regions have an average error of 0.480, compared with 0.258 for the remaining frames (as shown in Table 6). More importantly, 85.33% of the samples exhibit a local-to-rest error ratio of at least 1.5. This shows that large pose errors can be concentrated within a short gloss-associated region and therefore be less visible in errors averaged over the entire sequence.

Additional sensitivity-guided loss decomposition examples. Additional examples in Figure 9 show that our method produces softer allocations around transitions while keeping most supervision concentrated on nearby occurrences. Compared with CTC-based assignment, the transition between adjacent occurrences is less abrupt, especially when the corresponding motion changes gradually. We also observe that localization near the beginning or end of a sequence can be less precise, which may slightly shift subsequent occurrence assignments. These visualizations suggest that the proposed decomposition captures local temporal structure well in most cases, while remaining approximate around sequence boundaries.

Additional generation examples. As shown in Figure 10 and Figure 11, the generated poses on both CSL-Daily and Phoenix14T generally follow the ground-truth motion. Small errors can still be observed in some joint positions and local movements, but the main body motion and temporal changes are largely consistent with the reference sequences. These examples show that the model can produce stable pose sequences on both datasets while retaining the main motion content.

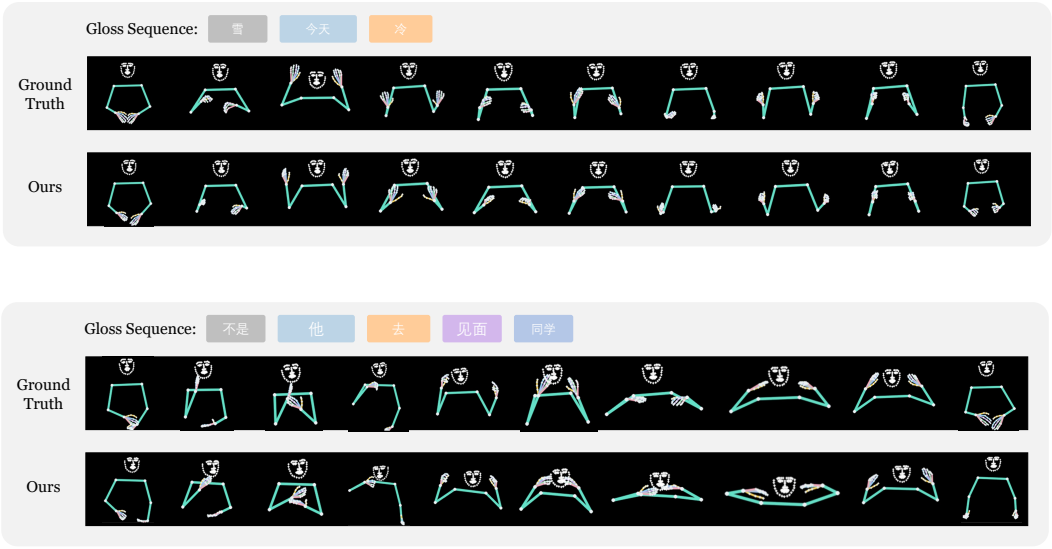


Figure 10: Additional G2P generation examples on CSL-Daily.

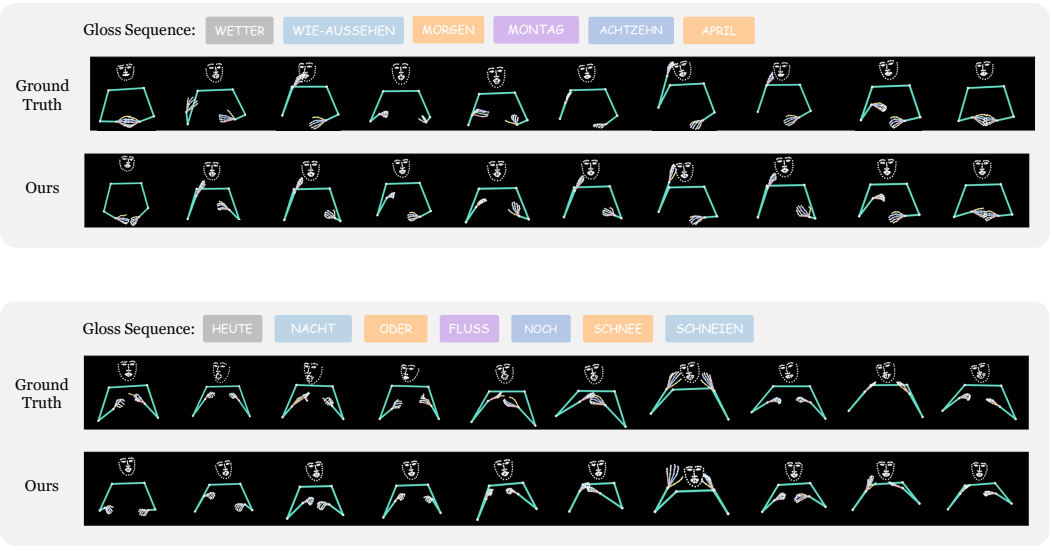


Figure 11: Additional G2P generation examples on Phoenix14T.