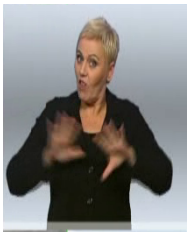
Modeling Intensification for Signed Language Generation: A Computational Approach

Anonymous ACL submission

Abstract

End-to-end sign language generation models do not accurately represent the prosody of the languages. This lack of temporal and spatial variation in generated signs leads to poor quality and lower human perception. In this paper, we seek to improve prosody in generated sign languages by modeling *intensification* in a data-driven manner with strategies grounded in the linguistics of sign language by enhancing the representation of intensity modifiers in gloss annotations. To employ our strategies, we first annotate a subset of the benchmark PHOENIX14T dataset with different levels of intensification. We then use a supervised intensity tagger to extend the tagging to the whole dataset. This enhanced dataset is then used to train state-of-the-art transformer models for sign language generation. We find that our efforts in intensification modeling yield better results when evaluated with automated metrics. Human evaluation also indicates a significantly higher preference of the videos generated using our strategies in the presence of intensity modifiers.[1]



`WOLKE` less clouds	`WOLKE` very cloudy
10 frames	17 frames
Low Reiteration	High Reiteration
No Delay	Delayed Beginning
Smaller Horizontal Space Use	Larger Space Use

Figure 1: In sign languages, modifiers are represented spatially and temporally and they change the semantics of the sign. Here, two signers from PHOENIX-14T manually sign German "less clouds", and "very cloudy". Both of these signs have the same gloss representation: `WOLKE` (cloud in German). They are figuratively the same sign, yet the duration, repetition, temporal pauses, and continuations determine the meaning. This information is lost during sign language translation and evaluation.

1 Introduction

Similar to spoken languages, signed languages have rich grammar rules and unique linguistic structures (Emmorey, 2001). Elements of prosody, such as rhythm, tempo, stress or lengthening play an important role in both spoken and signed languages (Brentari et al., 2018). Thus, it is important for sign language generation (SLG) systems to be able to model prosody. However, much of current study on prosodic markers such as intensification (Bolinger, 1972; Rett, 2008; Ghesquière and Davidse, 2011) are based on linguistic theories of spoken languages and cannot be adapted because prosody in sign language is represented in the visual modality (Wennerstrom, 2001). Spatial and temporal presentations such as iconicity, gesture duration, space utilization, as well as temporal pauses are used to stress on semantic differences (Wilbur et al., 2012). Due to such distinctive nature of sign language, the challenges of modeling prosody in SLG systems need to be addressed specifically.

Evidently, sign language generation (SLG) systems have been developing rapidly in recent years due to their potential importance to the Deaf and Hard of Hearing (DHH) communities (Stoll et al., 2018; Zelinka and Kanis, 2020; Stoll et al., 2020; Saunders et al., 2021). Transformer models (Saunders et al., 2020b) have been shown to outperform other neural models (Stoll et al., 2020) in generating sign language from gloss annotations —a shortened approximation of spoken language that

[1] We will make our annotated dataset and code publicly available upon paper acceptance.

has mapping to signs. One of the key limitations of state-of-the-art models is that the prosody of the sign videos generated by state-of-the-art models does not change with the semantics of the signs (Duarte et al., 2021). Given the recency of interest in the field, the problem of modeling prosody in sign language is yet to be tackled.

In this paper, we take a step toward the goal of modeling prosody in sign language generation by modeling *intensification*. We refer to intensification as the presence of *intensity modifiers* that quantify nouns, adjectives or adverbs in a sentence. The intensity modifiers can either be an amplifier (e.g., lot of rain) or a diminisher (e.g., little rain). Studies in the linguistics of sign languages show that intensity modifiers change the duration and tactile emphasis of the produced sign (Wilbur et al., 2012). Thus, intensification modeling can impact prosody of generated signs. However, this potential of intensification is not realized within current models because they depend on gloss representation. Intensity modifiers are often excluded in gloss representation because they are a sparse approximation of spoken language. As shown in Figure 1, the spatial and temporal properties of signs differ dramatically even when they map to the same gloss. State-of-the-art models cannot be aware of this temporal and spatial manipulation by modifiers if they are not represented in the gloss training data.

Our initial analysis of the PHOENIX-14T (Camgoz et al., 2018), a German Sign Language dataset, reveals that 23% of the data has at least one adjective or adverb in the text transcript but none in the gloss representation. Since adjectives and adverbs (e.g., little) often act as intensity modifiers, intensity modifiers are likely to be under-represented in the gloss as well. This observation motivates the need of explicit modeling of intensification in the gloss representation and modifying state-of-the-art models to incorporate this additional information. We hypothesize this to have an overall improvement in the models' performance both quantitatively in terms of automated metrics and qualitatively in terms of human evaluation. To this end, drawing on linguistics and cognitive science studies of sign languages, we make the following contributions in a data-driven way:

1. Introduction of gloss enhancement strategies grounded in linguistics that respect the differing information goals of modifiers with various levels of intensity.
2. Presenting a supervised tagging model to enhance a given gloss dataset with modifier intensity levels using strategies we identified.
3. Making available an enhanced version of the PHOENIX14T dataset where the glosses are tagged with intensity levels of modifiers.
4. Incorporating modifier information into the Progressive Transformer (PT) model. We also propose a novel model that can dynamically select the generated poses with different gloss enhancement as input. We make our code and data publicly available.[2]

2 Related Work

Prosody of Signed Languages Prosodic information in sign languages has been studied through the lenses of cognitive sciences and linguistics. Using brain images, Newman et al. (2010) show that prosodic signed information is processed in much the same way as it is in hearing speakers. In (Sandler, 1999), the intertwined nature of prosody is observed in a multifaceted manner for semantics, neurological basis and syntactic understanding of sign languages. Nicodemus et al., (2009) note that prosodic markers play an important role as delimiting units during the generation and perception of the signs.

In linguistics research, studies have focused on the relationship between prosody and syntax in sign language (Sandler, 2010), role of prosody in identifying break points in discourse and detection of salient events (Ormel and Crasborn, 2012). Sandler et al. (2020) suggest that pragmatic notions related to information structure are parts of prosody in sign languages. Although there has been limited work that highlight the importance of intensity modifiers in signed languages' prosody (Wilbur et al., 2012), our work is the first data-driven empirical study that studies a large dataset, then annotates, quantifies and characterizes data-driven strategies for modeling intensification. Moreover, none of the work cited in this subsection is computational. Our work is the first that presents a computational model for intensification as a step toward modeling prosody.

Sign Language Generation In contrast to the fields of cognitive sciences and linguistics, prosody is still unaddressed in the field of sign language generation (SLG). The primary aim of SLG is generating sign poses from texts. Earlier work has

[2]Data and model details are provided in the Appendix.

explored methods to generate animated avatars (Cox et al., 2002; Glauert et al., 2006; McDonald et al., 2015) from speech or text inputs, but were restricted by the rule-based systems and the modest size of sign pose libraries. More recently, with the introduction of large corpora such as PHOENIX14T (Camgoz et al., 2018) and How2sign (Duarte et al., 2021) and advanced deep learning model architectures, generating more accurate and expressive human skeletal sequences from spoken language transcripts or annotated glosses has become possible (Stoll et al., 2018, 2020; Zelinka and Kanis, 2020; Saunders et al., 2020a,b, 2021). Yet, none of these works attempt at modeling intensification or any other indicator of prosody. Our work is the first that combines linguistic and cognitive findings with computational models for the task of modeling intensification.

3 Intensification in Signed Languages

Gloss annotations in the German Sign Language weather forecast corpus, PHOENIX14T, are simple German words that often do not capture subtleties of sign language. For example, "very cloudy" and "slightly cloudy" are both represented by a single gloss "WOLKE" (CLOUD). Our analysis shows that in 23 percent of the data, the gloss representation does not contain any adjectives or adverbs present in the text transcript. Since intensity modifiers are usually adjectives/adverbs that quantify intensity of other words, we expect them to be missing from the gloss representation as well. Hence, in order for the model to represent intensity modifiers in its latent space, it is necessary to make them present in the training data.

3.1 Gloss Enhancement Strategies

We analyzed in a data-driven manner the best ways of representing intensity modifiers in gloss annotations based on the linguistic theories, cognitive science and neuroscience perspectives of intensities in signed languages. We discovered that the choice of order for the additional gloss modifier tokens matters. Linguistic analysis of American Sign Language also shows the importance of this.

Wilbur et al. (2012) explain that depending on the degree of the adjective, there is a "sharp movement to a stop" in the final timing of the sign, which is coined *end-marking*. They also show that the initial time interval of a sign also gets modified with a slight pause in the beginning and a faster continuation of the sign, which is termed as a *delayed-release*. Also, there exists other datasets with different annotation schemes, one of which –Public DGS Corpus– uses a gloss annotation convention where the phonemes and synonyms that have different signs contain a number that is added as a suffix to the end of the gloss (Konrad et al., 2020). Finally, as described by (Nicodemus et al., 2014) during the end-marking and elongation phase, a sign might be reiterated to mark the intensification.

Approach	**Example**
Text	very cloudy
Original Gloss	WOLKE (cloud)
Suffi.	WOLKE-INT2
End-mark.	WOLKE <INT2>
Delay.-rel.	<INT2> WOLKE
Suffix.-reiter.	WOLKE-INT2 WOLKE-INT2

Table 1: Gloss Enhancement examples.

Inspired by these previous work in linguistics of sign languages and in analyzing the dataset with sign language researchers, we came up with four strategies to better represent intensity modifiers in glosses. We use these strategies in four alternative ways, as shown in table 1 and are introduced below:

- **End-Marking**, where an additional token of <HIGH-INT> or <LOW-INT> is added *after* the intensity-modified gloss to represent the change in the final timing of the sign as shown in (Wilbur et al., 2012).
- **Delayed Release**, where the additional intensity modifier token of <HIGH-INT> or <LOW-INT> is added *before* the original gloss, as described in (Wilbur et al., 2012) to represent the delayed release in the initial timing of the sign.
- **Suffixation**, where an INT suffix is added at the end of the gloss with an additional numerical value (1 or 2) corresponding to the degree of intensification. This is analogous to the Public DGS Corpus annotation (Konrad et al., 2020).
- **Reiteration**, where we repeat the intensity-modified gloss token twice to capture this in the gloss representation as described by (Nicodemus et al., 2014).

3.2 Data Annotation

We start by selecting a subset of the publicly available PHOENIX14T dataset (Camgoz et al., 2018)

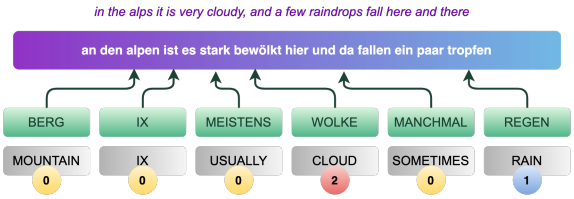


Figure 2: This figure shows an example annotation. German transcript text and gloss are provided as context along with their English translations. Each English gloss in the sentence are tagged with 0, 1, 2, corresponding to the degree of intensification.

Model	Features	Prec.	Recall	F1
SVM	W[2-5]	70.0	45.6	50.4
SVM	C[2-5]	63.8	54.0	57.2
FastText	embed	60.5	62.0	61.0
BiLSTM	embed	62.1	66.6	64.1
G-BERT	–	**74.3**	74.2	74.2
M-BERT	–	74.2	**76.4**	**75.3**

Table 2: GLOSS intensifier classification results. W stands for word, C stands for character. Embeddings for FastText and BiLSTM are learned during training.

for the annotations of intensity modification.

Data Sampling. Initial analysis demonstrated that gloss annotations tend to ignore the adjectives/adverbs, which are signals of intensity modification. We hypothesize that for samples where the number of adjectives/adverbs is zero in gloss annotations but more than zero in texts, the intensity information is more likely to be missed. We used Spacy (Honnibal and Montani, 2017) part-of-speech (POS) tagger to tag the text and gloss pairs, then utilize the hypothesis mentioned above to filter the data. In the end, we acquired 1557 samples in the train set, 132 samples in the development set, and 157 samples in the test set. Afterwards, the gloss sequences are split into individual gloss tokens. These gloss tokens are paired with the full text transcripts, which yields a total of 12.8K gloss token to sentence pairs – 10.8K from the 1557 instances in train, 1K from the 132 instances in dev and 1K from the 157 instances test set.

Annotation Protocol. For each of the gloss token to sentence pair, we ask at least one annotator to assign labels to the gloss token from the following categories: (i) 2 as "high intensity" if there is an intensity modifier such as "high" in the text surrounding the gloss; (ii) 1 as "low intensity" if the intensifier in the text marks a low degree intensity; or (iii) 0 if there is no corresponding modifiers in the text transcripts.[3] Figure 2 shows an example of the annotation.

Annotator Agreement. Three expert annotators were recruited according to the rules and regulations of our institution's human-subject board. Annotators were paid $15 per hour. To assess the inter-annotator agreement, we randomly sampled 700 token-sentence pairs and asked all three annotators to annotate. The resulting Fleiss' Kappa (Fleiss, 1974) coefficient is of 69.2, which suggests a substantial agreement among the annotators.

[3]We translated the German transcriptions and glosses into English using the Google Translate API https://cloud.google.com/translate

3.3 Full Corpus Intensity Enhancement

Utilizing the annotated pairs, we train a battery of classifiers to automatically predict the gloss labels for the remaining data points. Having an automated classifier saves us resources that would otherwise be needed to tag the whole dataset.

We frame the task as a text pair classification problem. Given the original text transcript and a gloss token, the goal is to predict a label from: "0" (no intensity modification), "1" (low degree intensity) and "2" (high degree intensity). We experimented with multiple classification baselines, including SVM with n-gram features, fastText (Joulin et al., 2017), Bidirectional LSTM and two versions of fine-tuned BERT (Devlin et al., 2019) models – German BERT (G-BERT) and multilingual BERT (M-BERT). All models are trained on the manually annotated 10.8K training pairs and results are reported on the 1K test subset.

Table 2 shows the experiments with different classifiers. Fine-tuned transformers G-BERT and M-BERT outperform others by a large margin. The performance improvement of M-BERT compared to G-BERT is statistically significant according to a permutation test.

We tag all the remaining glosses with the best-performing classifier, M-BERT, in the original PHOENIX-14T dataset. We end up with four version of enhanced gloss sequences by incorporating the aforementioned strategies in section §3, namely *Suffixation*, *End-marking*, *Delayed Release* and *Suffixation Reiterate*.

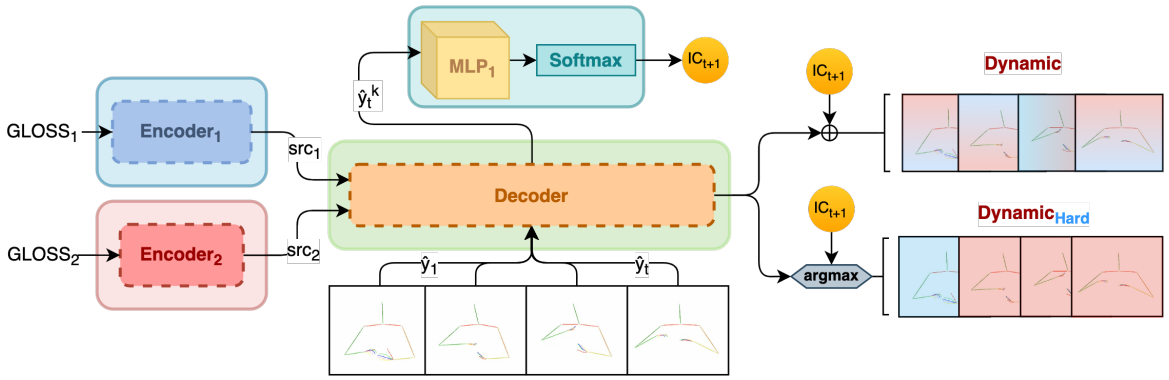


Figure 3: This figure shows the architecture of the Dynamic Selection model. The overall architecture is similar to the Progressive Transformer, except having two Encoders to select between two different types of strategies. MLP layer is the decisive step on selecting the strategy from the encoders. Dynamic model uses a weighted mixture of the decoder outputs (represented with a gradient of blue and red). Dynamic$_{hard}$ uses an argmax to pick a source.

4 Model

In this section, we first introduce a baseline model that has been widely adopted for the sign language generation task (section §4.1). To better model the signer's dynamic intensification choices during sign generation, we further propose a dynamic selection model (Figure 3) that makes use of inputs with different intensity modification strategies.

4.1 Progressive Transformer Baseline

The main goal of the sign language generation model is to transform a gloss or text sequence into skeletal pose coordinates per each frame of the signing video. Formally, given a gloss sequence $X = [x_1, ...x_N]$, a sign language generation model aims to learn the conditional probability $p = (Y|X)$ where Y represents the corresponding skeletal pose coordinate sequence $Y = [y_1, ...y_T]$. We use the Progressive Transformer (PT) (Saunders et al., 2020b) model as our baseline. The model employs an encoder-decoder architecture to produce a sign language sequence $\hat{Y} = [\hat{y}_1, ..., \hat{y}_T]$ in an auto-regressive manner. The encoder is composed of L transformer layers, each with one Multi-Head Attention (MHA) and a feed-forward layer. The computed representation of the source sequence is fed into a modified transformer decoder, which employs a counter-based decoding mechanism to guide the generation of continuous joint sequences $\hat{y}_{1:T}$ and deciding the end of the generated sequence. This decoding strategy can be formulated as below:

$$[\hat{y}_{t+1}, \hat{c}_{t+1}] = PT(\hat{y}_t|\hat{y}_{1:t-1}, x_{1:N}) \quad (1)$$

where $\hat{y}_{t+1}$ and $\hat{c}_{t+1}$ are the produced joint sequence and the counter value for the generated frame t+1. The model is trained using the mean square error (MSE) loss between the generated sequence $\hat{y}_{1:T}$ and the ground truth $y_{1:T}$:

$$L_{MSE} = \frac{1}{T}\sum_{i=1}^{T}(y_i - \hat{y}_i)^2 \quad (2)$$

It is worth noting that, as stated by (Huang et al., 2021), the proposed decoding mechanism provides weak supervisions with the initial ground-truth frame and guided counter sequences during the inference time.

4.2 Dynamic Selection Generator

The PT baselines can generate sign poses from a single source of gloss end-to-end. However, in different scenarios, the signers may employ diverse intensification strategies to present meanings for the same gloss word (i.e. they may use a gesture with a delayed-release to represent "heavy thunderstorm" and later employ an end-marking to strengthen the intensity of another sign). To model this, we propose a new structure on top of the PT baselines. Given a text sequence, we mix k sources of glosses with different information goals and generate sign languages that dynamically pick the source gloss. In general, we can have multiple encoders, $Encoder_{1...k}$, to encode the glosses separately and obtain the representations $src_{1...k}$. We utilize a single decoder to decode the output representation k times from k sources of encoders, each with a different encoded input representation:

$$src_k = Encoder(x^k_{1:N}) \quad (3)$$

$$\hat{y}^k_{t+1} = Decoder(\hat{y}^k_t|\hat{y}^k_{1:t-1}, src_k) \quad (4)$$

We employ a multi-layer perceptron (MLP) followed by a softmax activation function to produce

selection probability distributions of each source for individual frames, which we call as importance coefficients IC_{t+1}, that are conditioned on the decoded representations $\{\hat{y}^k_{t+1}\}$:

$$IC_{t+1} = \{\alpha^1_{t+1}, ..., \alpha^k_{t+1}\} = IC(\{\hat{y}^k_{t+1}\}) \quad (5)$$

This strategy is different from (Saunders et al., 2021) where our decoded representation y^k_{t+1} aims at generating source-dependent sequences, while (Saunders et al., 2021) applies the self-attention on the decoded sequences only. We have two variants while generating the weighted output: Dynamic and Dynamic$_{Hard}$. The final dynamic output is a weighted mixture of the two candidate sequences:

$$\hat{y}_{t+1} = \sum_{i=1}^{K} \alpha^k_{t+1} \hat{y}^k_{t+1} \quad (6)$$

In this specific model we set the k at 2. For the dynamic$_{hard}$ variant of the model which picks the most plausible view at each frame as $\hat{y}_{t+1} = \hat{y}^k_{t+1}$ where $k = \arg\max_i \{\alpha^i_{t+1}\}$.

5 Evaluations and Results

Evaluation of sign language generation is challenging due to the lack of an automated metric to assess the quality of generated signs. The standard practice (Saunders et al., 2020b) is to translate the poses back to text domain and compare with ground truth text. This is called back-translation. Such automatic evaluation however, cannot accurately capture the quality of the produced signs (Yin et al., 2021). Thus, to complement our automated evaluation, we ask sign language experts to evaluate the generated signs. Lastly, we perform a qualitative analysis of the back translated text to i) confirm increased presence of intensity modifiers, ii) identify limitations of our models, and iii) pitfalls of existing metrics.

5.1 Automatic Evaluation

Splits and Metrics. Prior analysis on a subset of the PHOENIX-14T's dev set unveils the imbalanced distribution of data regarding the intensity modification phenomena. Thus, results on the original data split could not faithfully evaluate the model's capability to generate intensification-specific sentences. To this end, we develop a new data split – we collect data points which have at least one gloss labeled as either low or high intensity to construct the "with intensification" subset, and leave the remaining in a "without intensification" group. We report the BLEU-1, BLEU-4 (Papineni et al., 2002), ROUGE (Lin, 2004) on the back translated texts. We retrain the Sign Language Transformer (Camgoz et al., 2020) (SLT) to translate the sign skeletal sequences back into German texts. For the more fine-grained settings of "intensification"-focused evaluation, we additionally report the BertScore (Zhang* et al., 2020), an automatic metric for text generation that correlates better with human judgements, to measure the semantic similarities. We report statistical significance with bootstrap resampling on both 90% and 95% confidence levels (Efron and Tibshirani, 1993; Koehn, 2004).

Result. We observe that, as shown in *full* columns of Table 3, the enhanced glosses improve the quality of skeleton generation on the original split of dataset. We can see that our proposed intensification enhancement techniques obtain an average of 0.6 improvement on BLEU-4 score over the dev set, with significant improvement of more than 1.6 on ROUGE. We do not observe significant difference on the test set evaluations. Our proposed models obtain the highest ROUGE score, with negligible drop of BLEU scores comparing to models based on single source of gloss on dev set.

Regarding the new "with" and "without intensification" splits, we first observe that there exists a considerable score difference across all three metrics between the two groups. We hypothesized that current sign language generation models are biased towards reconstructing sentences without any intensification modifiers and lack the capability to represent the intensity modification. Over the "with intensification" subset, most enhanced data obtain significant improvements on BLEU-1 and ROGUE score, which confirms that the intensity modifying strategies help preserve the semantic meanings. Meanwhile, *Suffixation* results in stable performance gain over the "without intensification" subset. This demonstrates the model's capability to distinguish between different intensified texts, such that the difference between *rain* and *shower* signs can be obtained while the provided glosses remain the same. The harnessing of repetitions on top of *Suffixation* glosses bring in minor improvements on "with intensification" dev cases, and major gains are attributed to the "without intensification" test cases. In the end, our proposed *Dynamic* model obtains the highest test set performance, where the

	with intensification (248)				without intensification (271)				full		
DEV SET											
	B_1	B_4	RG	BS	B_1	B_4	RG	BS	B_1	B_4	RG
Baseline	25.07	6.24	22.61	72.20	35.46	17.98	36.84	77.46	29.92	11.90	30.05
Suffix.	25.72	6.71	24.03**	72.61	37.73**	**19.35****	**38.92****	**77.88**	31.32*	**12.81**	31.81**
Delay.-rel.	27.03**	6.67	24.31**	72.97	**37.75****	18.39	38.55**	77.84	**32.03****	12.35	31.74**
End-mark.	**27.32****	**7.29**	24.46**	72.52	36.48	18.08	37.26	77.42	31.59*	12.51	31.15
Suff.-reiter.	26.23*	6.74	24.78**	72.78	35.98	17.97	37.92	77.74	30.77	12.20	31.64*
Dynamic	25.88	6.52	23.82*	72.54	35.65	17.80	37.59	77.86	30.44	11.99	31.01
$\text{Dynamic}_{\text{hard}}$	26.01	6.36	**24.98****	**73.06**	36.35	18.25	38.75**	77.87	30.83	12.20	**32.17****
TEST SET											
	with intensification (314)				without intensification (328)				full		
	B_1	B_4	RG	BS	B_1	B_4	RG	BS	B_1	B_4	RG
Baseline	25.28	5.92	21.98	72.02	35.17	17.40	35.97	76.85	29.86	11.51	29.13
Suffix.	26.31	6.54	24.56**	73.10	33.70	17.14	34.60	76.87	29.73	11.71	29.69
Delay.-rel.	19.33	3.43	16.29	69.56	**36.07**	17.53	36.49	77.31	27.08	10.27	26.61
End-mark.	23.98	6.67	22.38	72.09	34.94	17.28	35.27	76.60	29.05	11.73	28.96
Suff.-reiter.	25.04	6.24	23.41*	**73.13**	34.85	**17.63**	36.43	**77.65**	29.58	11.74	30.06
Dynamic	26.06	6.79	23.89**	72.76	35.42	17.21	**36.53**	77.42	**30.39**	11.79	**30.34**
$\text{Dynamic}_{\text{hard}}$	**26.51***	**6.95**	**24.68****	73.11	33.63	16.97	34.87	77.17	29.81	**11.81**	29.90

Table 3: Gloss to pose (G2P) model performances with different enhanced gloss as input. The original dev/test instances are split based on whether it contains tagged gloss produced by our best tagger in section §3.3. B_1, B_4, RG and BS refer to BLEU-1, BLEU-4, ROUGE and BERTScore respectively. The marks * and ** denote that the results are significant comparing to baseline with the significance level $p < 0.1$ and $p < 0.05$ respectively.

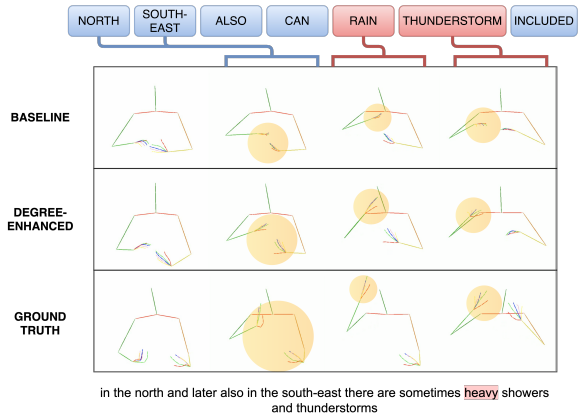


Figure 4: This figure illustrates the comparison between baseline and the intensification-enhanced model. Gloss annotations are linked to their corresponding frames. Here, ground truth skeleton uses wider movements due to the "heavy" modifier, and the intensification-enhanced outputs replicate the phenomena better than baseline.

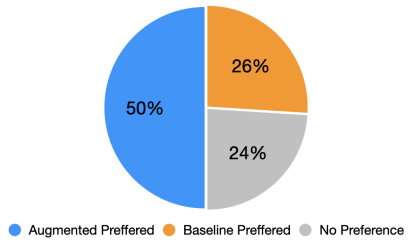


Figure 5: Human evaluation results for the generated skeletons.

gains are mainly attributed to the improvements over the "with intensification" subgroup.

5.2 Human Evaluation

We carried out a comparative human evaluation over 50 skeleton videos generated by both the baseline and our best performing model for human annotations. For each paired video, we asked deaf sign language users to identify the video that they found to be better than the other. They were specifically instructed to observe the following qualities and make their decisions: naturalness of the hand movements, alignment of the hand movements (excluding finger movements) with the ground truth, representation of intensity by the hand movements, and overall understandability.

As shown in Figure 5, outputs generated by our model trained on the enhanced glosses were preferred by signers (50% for our model vs. 26% for baseline). This difference is statistically different from chance as shown from a chi-squared test with $p = .00017$. This further suggests that a qualitative improvement using our enhancement strate-

	Examples (Translated from German)	**B_1**	**B_4**	**RG**	**BS**
Better capture of intensity modifiers					
G. Truth	The wind **usually** blows **weakly** from different directions.	-	-	-	
Baseline	The wind blows weak to moderate	47.8	0	55.7	81.9
Enhanced	The wind usually blows weakly from different directions.	100	100	100	100
Model hallucinations					
G. Truth	The wind blows **weak to moderate** at the sea also fresh	-	-	-	
Baseline	On the Alps and in the south, the wind blows weak to moderate	50	0	46.2	81.7
Enhanced	The wind blows in the south weak otherwise weak to moderately sometimes fresh to strong gusty from south to West	36.8	0	50.1	81.9
Metrics failure					
G. Truth	Tonight there are still **a few** thunderstorms possible in the south, otherwise rain only falls **here and there**, in places fog forms	-	-		
Baseline	Tonight, especially in the south and east there are rain or snow or freezing rain	37.9	15.4	39.6	75.4
Enhanced	Tonight, especially in the south and east here and there a few drops or flakes	32	0	36.9	75.6

Table 4: Examples of qualitative analysis over 100 back translated texts from the videos generated by baseline and our intensification enhanced model. **Bold** texts refer to the intensity modifiers that are missing in the gloss, blue highlight marks good generations and red highlight marks the errors. Our model can better retain the intensity information than the baseline. Meanwhile, as shown in the third example, n-grams based metrics may fail to reward the better intensity modifier representation.

gies is evident. Aspects that are not fully captured by the metric-based evaluations are more clear in the human evaluations which show that incorporating intensity into the model is crucial. Enhanced glosses can generate more natural videos that depict the intensity of the signs. It should be noted that the solution to the problem at hand needs further improvement as suggested by the considerable number of "no preference" votes.

5.3 Qualitative Analysis

We hypothesize that due to the inclusion of intensity modifiers in the gloss, there should be a higher presence of intensity modifiers in the back translated text. To verify this hypothesis, we compare the numbers of adjectives/adverbs in sentences back translated text from the baseline and the best model as an approximation of counting intensity modifiers. We observe more adjectives/adverbs (average of 3.42 comparing to baseline's 3.28) are being generated with the enhanced glosses.

To better understand our model's behavior, we manually inspect 100 instances randomly drawn from the "with intensification" cases for a qualitative analysis. We compare the back translated texts produced by the baseline and $Dynamic_{hard}$. The goal is not to evaluate overall quality of the back translated text but the presence and correctness of modifiers. The key observations are: i) in 30% of the cases, back translated text produced by our model has better representation of intensity modifiers compared to baseline, ii) in 3% of the cases, our model hallucinates and overproduces intensity modifiers, and iii) in 23% of the cases, at least two of the four automated metrics did not reward $Dynamic_{hard}$ for having better intensification. Table 4 shows examples of these observations.

6 Discussion and Conclusion

One limitation of our study is the lack of spatial and temporal context in the automated back-translation evaluation. The lack of a proper evaluation metric is a problem that needs to be addressed by an orchestrated effort from different fields surrounding the sign language research community. Another limitation is the cumulative error propagation that dissipates through the intensity classifier. then to the progressive transformer and then afterwards to the back-translation, amplifying total error.

Despite these limitations, we show that the strategies of intensification, grounded in the linguistics of sign languages, contribute to the improvement of end-to-end sign language generation systems. This modeling effort is supported by our metric-based and human evaluation results. We will make all data and code publicly available. For future work, we plan to further analyze the effects of these strategies on the perception of sign language understanding. We also plan to expand on the intensity modifier paradigm to further research in modeling prosody in sign language.

7 Ethical Considerations

Our work advocates for the need for more thoughtfulness of linguistic phenomena during the generation of sign videos. All models and analyses are built on a publicly available benchmarking dataset. We acknowledge that some modules of our model depend on pre-trained models such as word embeddings. These models are known to reproduce and even magnify societal bias present in their original training data (Li et al., 2021).

References

Dwight Bolinger. 1972. *Degree Words*. De Gruyter Mouton.

Diane Brentari, Joshua Falk, Anastasia Giannakidou, Annika Herrmann, Elisabeth Volk, and Markus Steinbach. 2018. Production and comprehension of prosodic markers in sign language imperatives. *Frontiers in Psychology*, 9:770.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7784–7793.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. 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*, pages 10023–10033.

Stephen Cox, Michael Lincoln, Judy Tryggvason, Melanie Nakisa, Mark Wells, Marcus Tutt, and Sanja Abbott. 2002. Tessa, a system to aid communication with deaf people. Assets '02, page 205–212, New York, NY, USA. Association for Computing Machinery.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Amanda Duarte, Shruti Palaskar, Lucas Ventura, Deepti Ghadiyaram, Kenneth DeHaan, Florian Metze, Jordi Torres, and Xavier Giro i Nieto. 2021. How2sign: A large-scale multimodal dataset for continuous american sign language.

Bradley Efron and Robert J. Tibshirani. 1993. *An Introduction to the Bootstrap*. Number 57 in Monographs on Statistics and Applied Probability. Chapman & Hall/CRC, Boca Raton, Florida, USA.

Karen Emmorey. 2001. *Language, Cognition, and the Brain Insights From Sign Language Research*. Psycology Press.

Joseph L. Fleiss. 1974. Statistical methods for rates and proportions.

Lobke Ghesquière and Kristin Davidse. 2011. The development of intensification scales in noun-intensifying uses of adjectives: sources, paths and mechanisms of change. *English Language and Linguistics*, 15(2):251–277.

John R. W. Glauert, Ralph Elliott, Stephen J. Cox, Judy Tryggvason, and Mary Christine Anne Sheard. 2006. Vanessa - a system for communication between deaf and hearing people. *Technology and Disability*, 18:207–216.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 249–256. PMLR.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Wencan Huang, Wenwen Pan, Zhou Zhao, and Qi Tian. 2021. *Towards Fast and High-Quality Sign Language Production*, page 3172–3181. Association for Computing Machinery.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431. Association for Computational Linguistics.

Diederik P Kingma and Jimmy Ba. 2015. Adam: a method for stochastic optimization. In *3rd International Conference on Learning Representations*.

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. Association for Computational Linguistics.

Reiner Konrad, Thomas Hanke, Gabriele Langer, Susanne König, Lutz König, Rie Nishio, and Anja Regen. 2020. Public DGS Corpus: Annotation Conventions / Öffentliches DGS-Korpus: Annotationskonventionen.

Luoqiu Li, Xiang Chen, Hongbin Ye, Zhen Bi, Shumin Deng, Ningyu Zhang, and Huajun Chen. 2021. On robustness and bias analysis of bert-based relation extraction. In *Knowledge Graph and Semantic Computing: Knowledge Graph Empowers New Infrastructure Construction*, pages 43–59. Springer Singapore.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81. Association for Computational Linguistics.

John C. McDonald, Rosalee J. Wolfe, Jerry Schnepp, Julie A. Hochgesang, Diana Gorman Jamrozik, Marie Stumbo, Larwan Berke, Melissa Bialek, and Farah Thomas. 2015. An automated technique for real-time production of lifelike animations of american sign language. *Universal Access in the Information Society*, 15:551–566.

Aaron J. Newman, Ted Supalla, Peter C. Hauser, Elissa L. Newport, and Daphne Bavelier. 2010. Prosodic and narrative processing in american sign language: An fmri study. *NeuroImage*, 52(2):669–676.

Brenda Nicodemus. 2009. *Prosodic markers and utterance boundaries in American sign language interpretation*. Gallaudet University Press.

Brenda Nicodemus, Laurie Swabey, and Christopher Moreland. 2014. *The Translation amp; Interpreting*, 6(1):1–22.

Ellen Onno Ormel and Ellen Onno Crasborn. 2012. Prosodic correlates of sentences in signed languages: A literature review and suggestions for new types of studies. *Sign Language Studies*, 12:279 – 315.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8024–8035.

J. Rett. 2008. Degree modification in natural language.

Wendy Sandler. 1999. Prosody in two natural language modalities *. *Language and Speech*, 42(2-3):127–142.

Wendy Sandler. 2010. Prosody and syntax in sign languages. *Transactions of the Philological Society. Philological Society*, 108 3:298–328.

Wendy Sandler, Diane C. Lillo-Martin, Svetlana Dachkovsky, and Ronice Müller de Quadros. 2020. Sign language prosody. *The Oxford Handbook of Language Prosody*.

Ben Saunders, Necati Cihan Camgöz, and R. Bowden. 2020a. Adversarial training for multi-channel sign language production. *ArXiv*, abs/2008.12405.

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

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

Stephanie Stoll, Necati Camgoz, Simon Hadfield, and Richard Bowden. 2020. Text2sign: Towards sign language production using neural machine translation and generative adversarial networks. *International Journal of Computer Vision*.

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

Ann Wennerstrom. 2001. *The music of everyday speech: Prosody and discourse analysis*. Oxford University Press.

Ronnie B. Wilbur, Evie Malaia, and Robin A. Shay. 2012. Degree modification and intensification in american sign language adjectives. In *Logic, Language and Meaning*, pages 92–101, Berlin, Heidelberg. Springer Berlin Heidelberg.

Kayo Yin, Amit Moryossef, Julie Hochgesang, Yoav Goldberg, and Malihe Alikhani. 2021. Including signed languages in natural language processing. 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 7347–7360, Online. Association for Computational Linguistics.

Jan Zelinka and Jakub Kanis. 2020. Neural sign language synthesis: Words are our glosses. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 3384–3392.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

A Gloss Classifier Implementation

SVM Baselines To construct the features for our text pair classification, we first concatenate the gloss token with the german text. Then we use term frequency-inverse document frequency (tf-idf) vectorizer to produce word and character n-gram vectors. These vectors are then used to train linear SVM classifiers. We use scikit-learn [4] implementation with default parameters for training. The SVM models primarily serve as baselines.

FastText In our implementation, we use two separate embedding layers. One for the text and one for the gloss token. The embeddings for the text is averaged using pooling and then concatenated with the embedding of gloss token. This concatenated vector is then passed through a linear layer and sigmoid function to produce the predictions. We use embedding size of 100 and train for 10 epochs. We cross-entropy loss and ADAM optimizer with default learning rate. We use PyTorch [5] for our implementation.

Bidirectional LSTM Similar to FastText, we have two separate embedding layers of size 100 for the text and the gloss token. the difference is that the output of text embedding layers are passed through a 2-layer bidirectional LSTM with hidden size of 300, dropout of 0.3. The output of the LSTM layers are then concatenated with the output of gloss embedding layer. The concatenated output is then passed through ReLU activation function and then passed through a linear layer. Similar to FastText, we train for 10 epochs, use cross-entropy loss and ADAM optimizer with default learning rate. PyTorch is used for implementation.

Fine-Tuned Transformers For our task. we fine-tune bert-base-multilingual (M-BERT) and german-bert-base-uncased (G-BERT) [6]. M-BERT is pretrained on Wikipedia text from 104 languages (including German). G-BERT is pretrained on Wikipedia dump, EU Bookshop corpus, Open Subtitles, CommonCrawl, ParaCrawl and News Crawl. The architecture of both models consists of 12 transformer blocks, hidden size of 768 and 12 self-attention heads. Since our task is classifying a pairs of texts, we fine-tune the models for sentence-pair classification. We use PyTorch implementation by HuggingFace [7] for the fine-tuning. We fine-tune for 5 epochs with learning rate of 5e-05.

Computational resources and running time Given our training data is small, the SVM baselines are very fast to train. They take less than 5 minutes to train. With an NVIDIA 2070 RTX GPU, the fastText and BiLSTM models take less than 10 minutes each. Fine-tuning each pre-trained BERT model with the same GPU but fewer epochs (5) take less than 10 minutes.

B Dataset Statistics

We use the publicly available benchmark, PHOENIX14T (Camgoz et al., 2018) dataset. This dataset comprises a collection of weather forecast videos in German Sign Language (DGS), segmented into sentences and accompanied by German transcripts from the news anchor and sign-gloss annotations. It contains videos of 9 different signers with 1066 different sign glosses and 2887 different German words. The video resolution is 210 by 260 pixels per frame and 30 frames per second. The dataset is partitioned into training, validation, and test set with 7,096, 519, and 642 sentences, respectively.

C Transformer (Re-)Implementation

We implemented Progressive Transformers models for sign language generation task (§4.1) based on the code [8] released by (Saunders et al., 2020b). Both encoder and decoder are built with 2 layers, 4 heads and embedding size of 256. We apply Gaussian noise with a noise rate of 5, as proposed by Saunders et al. (2020b). All parts of the network are trained with Xavier initialisation (Glorot and Bengio, 2010), Adam optimization (Kingma and Ba, 2015) with default parameters and a learning rate of 1e-3. The model takes 5 hours to train on 1 NVIDIA GeForce 1080Ti GPU. For our proposed Dynamic Selection model, both encoders and the decoder share the same settings as above. The Multi-Layer Percetron (MLP) model is composed of two linear layers with dimension of 1024 and a ReLU activation. The model takes 8 hours to train on 1 NVIDIA GeForce 1080Ti GPU. We implemented the back-translation model on top of

[4] https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html

[5] https://pytorch.org/

[6] https://huggingface.co/dbmdz/bert-base-german-uncased

[7] https://github.com/huggingface/transformers

[8] https://github.com/BenSaunders27/ProgressiveTransformersSLP

the original SLT code (Camgoz et al., 2020). The transformer models are built with 1 layer, 2 head and embedding size of 128. The feature size is changed to 150, which is the sequence length of generated skeleton joints sequence. The recognition loss weight and translation loss weight are set to 5 and 1 respectively. The model takes around 1 hour for training and evaluation. All models introduced above are implemented with Pytorch (Paszke et al., 2019).

D Retrained SLT model

Given the different versions of degree enhanced dataset (§3.3, we retrain the SLT models on the original text, skeleton joints sequence and the new gloss triples. This can serve as an estimation of the model's back translation quality given the oracle sign sequence. Table 5 shows the results.

	DEV SET					TEST SET				
Gloss Type	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE
Baseline	30.50	20.78	15.53	12.33	30.31	30.60	20.59	15.19	12.03	29.52
Suffix.	29.02	19.88	14.66	11.66	29.58	29.30	19.88	14.66	11.59	29.28
Delay.-rel.	28.72	19.71	14.79	11.77	29.63	29.31	19.93	14.70	11.62	28.98
End-mark.	29.28	19.99	14.99	12.01	29.88	29.32	20.01	15.01	11.93	29.04
Suffix. reiter.	31.15	21.80	16.50	13.14	31.11	29.76	20.77	15.70	12.60	29.15

Table 5: Translation results of the SLT model (Camgoz et al., 2020) used for back-translation. All models are trained and evaluated with ground truth hand and body skeleton joints (manual) and different choices of augmented gloss.