|  |  |
| --- | --- |
|  | **Hacettepe University**  Computer Engineering Department |

**Project Details**

|  |  |
| --- | --- |
| **Title** | Unsupervised Gloss-Free Sign Language Recognition Using Transformer Model |
| **Supervisor** | Assoc. Prof. Dr. Hacer Yalım Keleş |

**Group Members**

|  |  |  |
| --- | --- | --- |
|  | **Full Name** | **Student ID** |
| 1 | Eray Taymaz | 2210356123 |
| 2 | Tamer Yeşildağ | 2210356063 |

**Abstract of the Project**

|  |
| --- |
| Sign language is the primary mode of communication for many Deaf individuals, relying on complex visual gestures that pose unique challenges for automatic translation into spoken language. Traditional sign language translation systems heavily depend on gloss annotations, intermediate symbolic representations that require costly manual labeling and limit scalability. Recent research has explored gloss-free translation methods that eliminate the need for such annotations, enabling more flexible and data-efficient systems. Our project takes inspiration from the Sign2GPT framework, which leverages large pretrained vision and language models alongside a pseudo-gloss-based pretraining strategy for gloss-free sign language translation.  The problem we address is the development of a lightweight, gloss-free translation model that can be trained under constrained computational resources. We implement the core components of Sign2GPT’s architecture, including a frozen DinoV2 (ViT-S/14) vision backbone and a transformer-based spatio-temporal sign encoder. We pretrain this encoder to align sign video features with automatically extracted pseudo-glosses, using cosine similarity and binary classification loss. Due to VRAM limitations on Google Colab, we were unable to incorporate Low-Rank Adapters (LoRA) as in the original paper, and instead focused on optimizing a smaller model architecture. We also experimented with an alternative vision backbone (vit\_small\_patch32\_224), but it underperformed relative to DinoV2.  Training was conducted using augmented datasets with frame subsampling and color jittering, though augmentation did not improve downstream performance as expected. Our best model achieved modest results in pseudo-gloss prediction (F1 ≈ 0.25), indicating some learning of semantic alignment despite limited compute and training time.  The project demonstrates that pseudo-gloss-based pretraining can yield interpretable sign features even without gloss annotations or full model fine-tuning. While performance was limited by hardware constraints and reduced training time, the results validate the feasibility of gloss-free sign language understanding in low-resource settings. Future work should focus on enabling LoRA adaptation under memory constraints, exploring more robust augmentation pipelines, and extending the system to support full sign-to-text translation using pretrained language decoders. Overall, this project contributes a practical foundation for developing scalable and accessible sign language translation tools. |

**Introduction, Problem Definition & Literature Review**

|  |
| --- |
| 1. Introduction and Problem Definition Sign languages are complex visual languages utilized by Deaf communities for daily communication. Translating sign language into spoken language presents a multifaceted computational challenge, requiring the integration of computer vision and natural language processing (NLP). Traditional approaches to automatic Sign Language Translation (SLT) have depended heavily on gloss annotations, intermediate symbolic representations of signs aligned in a sentence-like order. However, these annotations are labor-intensive to generate and limit scalability across different sign languages and datasets.  In this context, gloss-free sign language translation has emerged as a critical research area aimed at bypassing manual gloss annotations while still achieving high-quality translation. The objective of this project is to develop a gloss-free SLT system inspired by the Sign2GPT architecture, and to explore practical constraints in deploying such a system under limited computational resources. 2. Literature Review Traditional SLT methods have relied on gloss-based pipelines, using continuous sign language recognition (CSLR) modules trained with Connectionist Temporal Classification (CTC) losses to identify gloss sequences from sign video frames. These methods achieve strong performance but incur substantial data annotation overhead.  Recent efforts have shifted towards gloss-free approaches to avoid manual gloss alignment. One such method, Sign2GPT, proposes a novel architecture that leverages large pretrained vision and language models, using Low-Rank Adapters (LoRA) for fine-tuning within memory constraints. The framework introduces pseudo-gloss pretraining, where pseudo-labels are derived automatically from spoken sentence lemmatizations and used to guide the visual encoder. The system integrates frozen DinoV2 vision features with a lightweight transformer encoder and a frozen multilingual GPT-like decoder (XGLM), with cross-modal alignment performed through gated attention mechanisms.  While this model significantly narrows the gap between gloss-based and gloss-free performance, its adoption in practice is limited by its dependency on high-end hardware (e.g., A100 GPUs) and the complexity of LoRA injection mechanisms. Additionally, attempts to generalize to different vision backbones such as ViT-S/32 or CLIP often lead to decreased accuracy due to training-incompatibility with sign language’s unique spatiotemporal characteristics. 3. Hypothesis and Research Objectives Our primary research question is:  Can a simplified and resource-constrained implementation of the Sign2GPT framework still learn meaningful pseudo-gloss representations from sign language videos to enable downstream translation, despite limited VRAM and compute capacity?  Based on this, our hypothesis is that pseudo-gloss supervision alone can provide significant representational grounding to a lightweight transformer encoder, even in the absence of LoRA adapters or full fine-tuning of the vision backbone. 4. Proposed Solution and Contributions To explore gloss-free sign language recognition under limited computational resources, we developed a simplified version of the Sign2GPT framework. Our model uses a frozen DINOv2 vision transformer as a spatial feature extractor and a lightweight transformer encoder to process temporal information. Instead of LoRA-based fine-tuning (which proved infeasible due to memory constraints) we focused on aligning visual features with pseudo-gloss prototypes derived from sentence transcripts using cosine similarity and binary cross-entropy loss. We also experimented with an alternative vision backbone (vit\_small\_patch32\_224), though it underperformed relative to DINOv2. Data augmentation techniques, including color jitter and random cropping, were used to expand the dataset size, but ultimately did not improve performance. Despite these limitations, our model was able to learn some degree of semantic alignment, validating the potential of pseudo-gloss-based pretraining as a lightweight strategy for gloss-free sign language processing. |

**Methodology**

|  |
| --- |
| 1. Dataset and Preprocessing1.1 Dataset The dataset utilized in this project consists of sign language video samples derived from RWTH-PHOENIX-Weather 2014T, a standard benchmark dataset in sign language translation. The dataset includes video clips of German sign language (DGS) paired with spoken German sentence translations.   * Total Samples: 7,096 sign language videos * Labels: Binary vectors representing the presence or absence of pseudo-glosses extracted from the corresponding spoken language sentences  1.2 Preprocessing The following preprocessing steps were applied to the dataset:   * Frame Subsampling: Every second frame was retained to reduce temporal redundancy and computational load, following the strategy used in Sign2GPT. * Pseudo-Gloss Extraction: Spoken language sentences were processed using spaCy's lemmatization and POS-tagging pipelines. Only content-bearing words with POS tags in {"NOUN", "NUM", "ADV", "PRON", "PROPN", "ADJ", "VERB"} were retained. This resulted in pseudo-gloss sequences representing the salient semantic elements of each sentence. * Data Augmentation: Each frame was subjected to random color jitter and resized cropping. These operations were applied consistently across all frames in a sequence. Although three variants per video were created, empirical evaluation suggested that augmentation negatively impacted model performance. * Masks: Padding masks were generated to handle sequences of varying lengths during batch processing.   2. Feature Extraction and Representation 2.1 Spatial Feature Backbone A frozen DINOv2 Vision Transformer was used as the visual feature extractor. The model outputs a 384-dimensional embedding per frame using the class token representation. Due to computational constraints, Low-Rank Adapters (LoRA) were not successfully integrated into the spatial backbone. 2.2 Input Projection A linear layer was applied to map the DINOv2 output to the transformer hidden size. 384-dimensional input embedding, which maps the input to a hidden dimension d = 300 or d = 512.  3. Sign Encoder Architecture 3.1 Positional Encoding Temporal information is injected using sinusoidal positional encodings. These embeddings are added to the input sequence to maintain relative position awareness. 3.2 Transformer Encoder The core model is a 3 to 4-layer Transformer encoder with the following properties:   * Feedforward size = 1024 or 2048 * Hidden size = 300 or 512 * 6 to 8 attention heads * Dropout = 0.1 to 0.2  3.5 Output Projection An optional output projection layer maps hidden states back to 300-dimensional space, aligning with prototype embeddings.  4. Pseudo-Gloss Pretraining 4.1 Learnable Prototypes A matrix of learnable prototypes, where the number of pseudo-glosses is initialized from pretrained fastText embeddings and updated during training. 4.2 Dual Softmax Matching For each encoded sign frame, cosine similarity is computed with all prototypes:  Softmax is applied over both temporal (t) and prototype (u) axes using learnable temperature parameters:    The final localization matrix is:  The predicted probability of each prototype’s presence is:    The loss is the binary cross-entropy between predicted and true pseudo-gloss presence:    5. Training Setup 5.1 Optimization  * Optimizer: Adam * Learning Rate: 3e-4 * Scheduler: OneCycleLR with 5% warmup * Batch Size: 2 to 8 (adjusted for memory constraints) * Gradient Clipping: * Epochs: 100  5.2 Hardware Constraints Training was conducted on Google Colab with A100 GPUs, encountering out of memory errors during attempts to include LoRA adapters. This constraint limited architectural complexity and batch size.  6. Evaluation Principles  The model was evaluated in the multi-label classification setting. Each video was expected to activate one or more pseudo-gloss prototypes. The following metrics were used:   * Precision (micro): Fraction of correctly predicted prototypes among all predicted * Recall (micro): Fraction of correctly predicted prototypes among all actual positives * F1 Score (micro): Harmonic mean of precision and recall * Thresholding: Probabilities were thresholded at 0.2 to determine binary activation   These metrics reflect the ability of the sign encoder to localize and identify meaningful pseudo-gloss representations without direct gloss supervision. |

**Results & Discussion**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1. Training and Validation Dynamics1.1 Training Regimes Two training strategies were employed:   * Constant learning rate: All experiments used a fixed small learning rate. * Scheduled learning rate: Experiments used a OneCycle scheduler with warm-up and a maximum learning rate.  1.2 Optimization Observations The constant learning rate strategy led to smoother convergence in terms of F1 score, and higher peak values compared to scheduled LR. The more robust and simple models are scored higher than complex models, meaning that more complex it gets more overfitting issues are happening due to using frozen DINOv2. Scores for example:   * With constant LR and augmentation, a 3-layer, 1024-FF model reached F1 ≈ 0.2175. * Scheduler-trained variants with the same config achieved peak F1 ≈ 0.1674 when augmentation was used. * With constant LR and no augmentation, the same model reached F1 ≈ 0.2845.  2. Performance Evaluation2.1 Metrics All models were evaluated on the pseudo-gloss presence prediction task using:   * Precision, Recall, and F1 Score   Predictions were thresholded at 0.2, consistent with Sign2GPT’s gloss-free evaluation protocol.   |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Layers** | **FF Size** | **Hidden** | **Dropout** | **Heads** | **Params** | **F1 Score** | **Augmentation** | **LR Type** | **Model** | | 2 | 256 | 300 | 0.2 | 3 | ~1.8M | 0.2147 | No | Constant | DINOv2 | | 3 | 1024 | 300 | 0.1 | 6 | ~3.7M | 0.2845 | No | Constant | DINOv2 | | 2 | 1200 | 300 | 0.2 | 2 | ~3M | 0.2804 | No | Constant | DINOv2 | | 3 | 1024 | 300 | 0.1 | 6 | ~3.7M | 0.2175 | Yes | Constant | DINOv2 | | 2 | 1200 | 300 | 0.2 | 2 | ~3M | 0.1671 | Yes | Constant | DINOv2 | | 3 | 1024 | 300 | 0.1 | 6 | ~3.7M | 0.1674 | Yes | Scheduler | DINOv2 | | 3 | 1024 | 512 | 0.1 | 8 | ~7.3M | - | No | Scheduler | DINOv2 | | 4 | 2048 | 512 | 0.2 | 8 | ~13.7M | - | Yes | Scheduler | DINOv2 | | 3 | 1024 | 300 | 0.1 | 6 | ~3.7M | - | Yes | Scheduler | ViT-S/32 | | 3 | 1024 | 512 | 0.1 | 8 | ~7.3M | - | No | Scheduler | ViT-S/32 | | 3 | 1024 | 300 | 0.1 | 6 | ~3.7M | 0.1255 | No | Constant | ViT-S/32 |  |  | | --- | | Interpretation: |  * 3-layer and higher feedforward widths (≥1024) produced the strongest performance. * Augmentation degraded results consistently across all training schedules. * The reason some models do not have F1 score is models loss graph indicates zero learning that testing is considered unnecessary.  3. Comparison with Sign2GPT The original Sign2GPT model used LoRA-injected DINOv2 and achieved BLEU-4 scores exceeding 23. Our implementation omits LoRA, focusing purely on pretraining a spatiotemporal encoder with pseudo-gloss targets.  Nonetheless, F1 scores in the 0.25–0.28 range (e.g., 0.2845 peak) suggest meaningful pseudo-gloss grounding in visual modality alone. While direct comparison to BLEU-based translation is not feasible, our results form a viable pretext task for future SLT modules. 4. Backbone Comparison: DINOv2 vs. ViT-S/32 One of the most critical architectural choices in this project was the selection of the frozen model. While DINOv2 served as the primary backbone throughout most experiments, experiments also included ViT-Small (ViT-S/32, patch size 32x32) trained on ImageNet-21k, available as vit\_small\_patch32\_224 from the torchvision.models module.  These models were compared using identical Transformer encoder settings (e.g., 3L-512-300). When evaluating performance:   * ViT-S/32 backbone reached a maximum F1 of 0.1255 * DINOv2 (frozen) achieved F1 scores in the 0.25–0.28 range with the same encoder setup   The performance gap reflects DINOv2's stronger representation quality for visual semantics relevant to human actions, gestures, and handshapes. 5. Ablation Insights5.1 Effect of Augmentation Augmentation hurt model performance:   * F1 dropped by ~5–7% when using color jitter and crop-based transformations. * These likely obscure fine-grained hand and facial motions crucial to sign semantics.  5.2 Depth, Width, and Positional Encoding  * 3-layer encoders with 1024 feedforward size offered the best trade-off. * Larger hidden widths (512) and deeper networks (4 layers) could not be trained to convergence under current constraints.  6. Localization Matrices and Qualitative Analysis In addition to numerical evaluations, qualitative visualization of the model's temporal alignment behavior was performed using localization matrices. These matrices indicate pseudo-gloss activation confidence across video time steps. Two representative test sequences are shown in the project poster (Figures 1 and 2), providing insight into the model's interpretability and grounding effectiveness.  In Figure 1, corresponding to the sentence "und nun die wettervorhersage für morgen mittwoch den dreißigsten märz", the model displays sharp activations for "wettervorhersage" at the beginning of the video, followed by sequential peaks aligned with "nun," "mittwoch," "dreißigsten," and "märz." This alignment indicates the encoder’s ability to temporally localize semantically relevant content without explicit supervision.  In Figure 2, for the sentence "der wind weht schwach bis mäßig aus süd bis südwest", localized activation is observed for "wind," "schwach," and "mäßig," with lower yet coherent response for "südwest." These results suggest the model’s effectiveness in detecting gloss-relevant visual features across variable sentence structure and signer style.  Such visualizations serve as a form of weak interpretability, offering evidence that the model does not merely memorize prototypes but actively learns to associate temporal video segments with semantic units. This insight supports the hypothesis that gloss-free training with pseudo-labels can yield semantically meaningful grounding, even in a weakly supervised setting. |

**The Impact and Future Directions**

|  |
| --- |
| The outcomes of this project have significant implications for the development of lightweight, gloss-free sign language processing systems. Traditional sign language translation (SLT) pipelines rely heavily on gloss annotations, which are expensive and time-consuming to obtain at scale. By contrast, this work demonstrates that meaningful pseudo-gloss representations can be learned from visual-only data using a relatively compact encoder architecture trained with weak supervision.  In practical terms, the proposed encoder design can be integrated as a pretraining module for downstream SLT systems or sign-to-text generation frameworks, especially in low-resource settings where gloss annotations are unavailable. Additionally, its lightweight nature (with as few as 1.8 to 3.7 million parameters in viable models) makes it suitable for deployment in edge devices or mobile applications intended to provide sign language assistance in real time. For example, real-time captioning services, educational tools for deaf students, or accessibility tools embedded in consumer devices could benefit from a modular, gloss-free sign representation layer.  From a research standpoint, this project contributes to an emerging body of work that seeks to bridge vision and language in sign language using proxy tasks. The pseudo-gloss prediction objective, borrowed and adapted from Sign2GPT, is shown to be a useful self-supervised signal even without the full translation stack. This result invites further exploration of weak or self-supervised signals in the broader sign language domain, particularly for underrepresented languages.  If continued in the future, the project could advance along multiple dimensions. Fine-tuning with a translation head, potentially even adopting the decoder module from Sign2GPT, would allow end-to-end evaluation of translation quality. Deeper transformer stacks could be trained using LoRA or QLoRA optimizations, which would help manage GPU memory costs while scaling model capacity. Performance could also benefit from more sophisticated data augmentations that preserve spatial integrity or contrastive learning methods that align sign segments temporally.  Moreover, extending the system to multilingual sign corpora and integrating gesture-to-speech output pathways would increase its inclusivity and social impact. With additional development and broader dataset access, this encoder could serve as a general-purpose visual language model for sign language, eventually contributing to open-source toolkits or being submitted to relevant academic venues such as ECCV and CVPR |