Leveraging Phonological Clustering for Word-Level Bangla Sign Language Recognition

Anonymous ACL submission

Abstract

Bangla Sign Language (BdSL) recognition faces challenges from signer variability and spatiotemporal complexity. Conventional flat classifiers often ignore phonological relationships that can enhance recognition accuracy. We propose a hierarchical framework that integrates confusion-driven phonological clustering into a Bidirectional Long Short-Term Memory (Bi-LSTM)–based pipeline. Baseline classification using five Bi-LSTM variants establishes accuracy–complexity trade-offs, followed by clustering of visually and phonologically similar signs based on confusion patterns. Cluster-specific feature engineering and retraining further refine recognition. Experiments are conducted on a curated 50-class subset of the SignBD-Word dataset. In the baseline setting, the most complex model (Bi-LSTM-1) achieves 89.72% accuracy with 1.05 million parameters. With the proposed confusion-matrix driven clustering architecture, five out of six clusters employing the lightweight Bi-LSTM classifiers outperform the baseline model. Specifically, Bi-LSTM-1 reaches a weighted average accuracy of 93.32%, Bi-LSTM-2 achieves 92.40%, Bi-LSTM-3 attains 92.24%, and Bi-LSTM-4—using only 140K parameters (13.3% of Bi-LSTM-1)—achieves 91.12%. Notably, the lighter models maintain strong cluster-level performance while significantly reducing model size, with Bi-LSTM-4 achieving an 86.7% reduction in parameters. This demonstrates that confusion-matrix-based phonological clustering enables lightweight models to achieve competitive performance at both cluster and overall dataset levels in large-vocabulary sign language recognition tasks.

1 Introduction

Sign language is a complete and expressive visual language that conveys meaning through a combination of hand gestures, facial expressions, and body movements. As the demand for inclusive technologies grows, automatic Sign Language Recognition (SLR) systems have emerged as vital tools for bridging the communication gap between signing and non-signing communities. SLR research typically spans various levels of linguistic granularity, including fingerspelling, word-level, and sentence-level recognition. In this study, we focus specifically on video-based, word-level recognition in BdSL.

Despite growing efforts in BdSL recognition, most existing approaches adopt flat, single-stage classifiers that map keypoint sequences directly to sign labels. These models often treat each sign independently, ignoring deeper phonological or visual relationships between signs—especially problematic when visually similar signs differ only in subtle hand movements or temporal dynamics. To overcome these challenges, we propose a hierarchical recognition framework based on confusion-matrix-driven clustering. The evaluation explores three key aspects: (i) accuracy across five Bi-LSTM configurations, (ii) improvements in separating visually similar signs, and (iii) the impact of clustering on reducing model complexity without sacrificing performance. This study ultimately demonstrates that confusion-informed clustering enables the use of lightweight models while maintaining or even improving recognition accuracy.

The paper is organized as follows: Section 2 reviews related work, Section 3 describes methodology, Section 4 presents experimental results, and Section 5 concludes with key findings. Limitations are also discussed.

2 Related Work

Research in Bangla Sign Language (BdSL) has evolved from static image-based methods to pose-driven deep learning approaches leveraging spatiotemporal keypoints.

2.1 Keypoint-based Recognition in BdSL

(Hasan and Adnan, 2025) introduced EMPATH, a framework that integrates ensemble learning, MediaPipe Holistic features, and attention-based Transformers for BdSL recognition. The model achieved state-of-the-art results of 79.81% on SignBD-Word-90 and 70.58% on SignBD-Word, outperforming previous baselines. A key innovation was its interpolation module, designed to mitigate missing hand landmarks, which improved robustness in real-world conditions.

The BdSLW60 dataset (Rubaiyeat et al., 2025) contains 9,307 videos across 60 signs from 18 signers. Using 543 landmarks per frame, an attention-based Bi-LSTM achieved 75.1% accuracy, showing the effectiveness of spatio-temporal models in noisy settings. Similarly, Das et al. (2025) proposed a deep LSTM trained on 15,000 videos spanning 100 signs. Their 4-layer model reached 88.33% accuracy and 0.9856 AUC, though it was collected under controlled conditions without signer diversity.

Multi-view recognition has also been explored. The MVBSL-W50 dataset (Islam et al., 2024) provides 4,000 videos of 50 signs captured from frontal, left, and right viewpoints. An attention-based Bi-GRU trained on this dataset achieved 89.69% accuracy. In another large-scale study, Shakil et al. (2024) used 8,790 videos (135 signs) with 1,662 MediaPipe landmarks per frame to train a CNN-LSTM, reaching 96.82% test accuracy. At the phrase level, Mahin et al. (2023) proposed a 3-layer LSTM trained on 1,134 samples covering 27 phrases. While achieving 92.07% accuracy, the model suffered from false positives due to frame-wise predictions at fixed intervals.

2.2 Clustering-based Approaches in Related Domains

To date, no BdSL research has explicitly explored clustering strategies to address inter-class similarity. In speech recognition, Oh et al. (2021) proposed hierarchical phoneme classification using confusion-matrix-based clustering on the TIMIT dataset. Training specialized Bi-LSTMs per group improved accuracy by 2.2% overall and up to 6.0% for acoustically similar phoneme classes.

In sign language, Phothiwetchakun and Rakthanmanon (2021) applied hierarchical clustering for Thai fingerspelling recognition. Hand landmark features were grouped into clusters of similar gestures, and within-group classifiers achieved up to 86.71% accuracy—outperforming single-level models, though limited to static signs.

2.3 Summary

Overall, BdSL research demonstrates strong progress in leveraging spatiotemporal landmarks and attention mechanisms.Yet, existing models still struggle with visually similar signs, and clustering-based methods remain unexplored in this context. Motivated by successes in speech and other sign languages, this work investigates confusion-matrix-driven clustering to improve recognition of BdSL signs.

3 Methodology

This section presents the methodology of this work, outlining a hierarchical recognition framework for BdSL. Figure 1 outlines the complete pipeline, illustrating how clustering and cluster-specific modeling are leveraged to improve recognition performance.

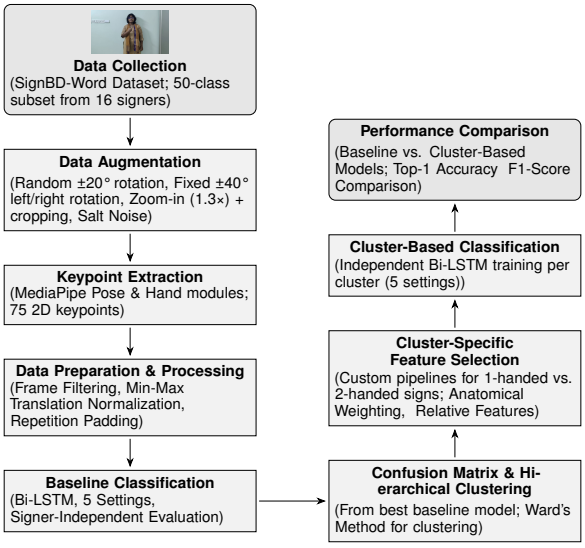

Figure 1: Methodology pipeline for phonological clustering-enhanced BdSL recognition.

The process involves a multi-stage pipeline beginning with video data collection, augmentation, and phonological feature extraction using MediaPipe(Lugaresi et al., 2019). These features feed into a series of Bi-LSTM models trained for baseline recognition. Confusion matrix analysis then drives hierarchical clustering of visually or semantically similar signs, followed by cluster-specific modeling. Each of these stages is detailed in the

subsequent sections.

3.1 Data Collection

This study utilizes a 50-class subset of the SignBD-Word dataset (Sams et al., 2023), a large-scale video corpus for BdSL recognition. The selected subset includes 24 one-handed and 26 two-handed signs, chosen randomly. Each word is represented by 30 video samples captured in both full-body and upper-body views from 16 signers. The original dataset was recorded using high-resolution (1920×1080) cameras under varied lighting and background conditions to simulate real-world variability. The selected 50 signs used in this study consist of both one-handed and two-handed gestures, categorized as follows:

One-Handed Signs (24): aam, aat, abhawa, akash, amader, amar, ami, amra, apnader, apnara, baba, bah, bangladesh, batash, bathroom, bhul, boka, bon, brihospotibar, budhbar, chacha, chachi, chagol, chalak

Two-Handed Signs (26): boro, boi, bichana, bagan, ador, agun, attio, baari, chabi, dorja, doctor, football, goru, glass, golarhaar, hata, hospital, chata, chih, chikitsha, chocolate, choto, cricket, dengue, durbol, janala

3.2 Data Augmentation

To improve generalization across signer variability, video-level augmentation was applied using random $\pm 20^\circ$ rotations and salt noise, initially doubling the dataset to 60 videos per class. Further augmentations included fixed left and right $\pm 40^\circ$ rotations, zoom-in (1.3×) with signer-centered cropping, and salt noise. The rotation angles (±20° and ±40°) and zoom factor were empirically chosen to simulate natural signer variations without introducing excessive distortion that could hinder recognition. Implemented using MoviePy, scikit-image, and OpenCV(Team, 2024), this multi-stage strategy significantly boosted dataset diversity and robustness for training.

3.3 Keypoint Extraction

We employed the MediaPipe framework (Lugaresi et al., 2019) to extract 2D anatomical landmarks from each video frame. Using both the pose and hand modules, up to 75 keypoints were obtained per frame—33 from the body and 21 from each hand—capturing spatial information for critical regions such as the shoulders, elbows, wrists, torso, and individual finger joints. The resulting (x, y) coordinates form the foundation for computing phonological features like hand location, orientation, and movement.

3.4 Data Preparation & Processing

After extraction, a cleaning step was applied to improve the integrity of the data: frames in which both hands were undetected were discarded, as these typically resulted from motion blur, occlusion, or tracking failure. This filtering ensured that only reliable and informative keypoints were retained for subsequent processing. To minimize signer-specific variability and ensure consistency across frames, we applied a two-stage normalization process. First, min-max normalization was used to rescale the 2D keypoints to a $[0, 1]$ range for each frame. Then, translation was applied to shift a reference keypoint (the nose) to the origin, effectively centering the signer in the coordinate space. This normalization strategy follows established practices in (Schneider et al., 2019) and enables the model to focus on gesture dynamics rather than signer position or frame size.

The normalization was applied independently to each axis using the following equation:

$$x' = \frac{x - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}} \tag{1}$$

where x is the original coordinate value, and x_{min}, x_{max} are the minimum and maximum values across all joints in that frame. Finally, since sequence lengths varied after motion-based frame selection and cleaning, we adopted a repetition-based padding strategy to standardize input dimensions. Each sequence was extended to match the maximum observed length (max_frames) by cyclically repeating existing frames, while sequences longer than this threshold were truncated. This method, inspired by the "sign padding" approach in (Gil-Martín et al., 2023), preserved the temporal rhythm of the original sign and avoided the discontinuity commonly introduced by zero-padding. As a result, the processing pipeline ensured that the input fed into the classification model was clean, normalized, and temporally aligned—contributing to more stable and accurate model training.

3.4.1 Model Configuration

To investigate the trade-off between model capacity and computational efficiency in BdSL recognition,

we designed five Bi-LSTM configurations with systematically reduced complexity. Each model follows the unified architectural framework shown in Figure 2, while varying in hidden size, layer depth, attention dimension, and total parameter count. The complete set of configurations is presented in Table 1.

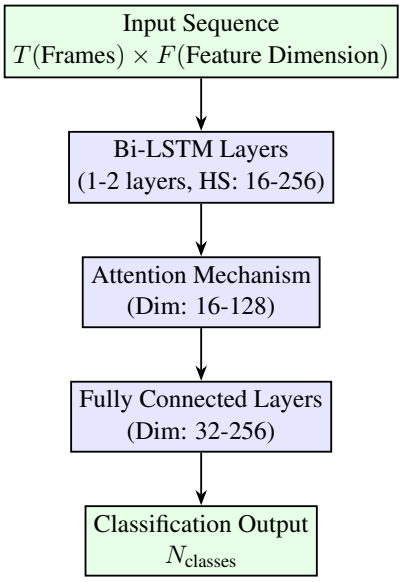

Figure 2: Architectural framework for Bi-LSTM-based BdSL recognition.

Core Architectural Components: Each model processes an input $X \in \mathbb{R}^{T \times F}$, where T is the number of frames and F the feature dimension. The backbone consists of 1–2 Bi-LSTM layers (hidden size 16–256) that capture temporal dependencies. Their outputs are passed through a temporal attention layer, which highlights the most discriminative frames. Finally, a two-layer multi-layer perceptron (MLP) with BatchNorm, ReLU, and Dropout maps the attended representation to the N sign classes.

Controlled Experimental Design: The five Bi-LSTM configurations in Table 1 were designed to systematically assess the trade-off between accuracy and efficiency. Bi-LSTM-1 and Bi-LSTM-2 enable a direct comparison of network width versus depth. Model 3-5 form a progression of reduced hidden sizes, allowing evaluation of how decreasing model capacity impacts recognition performance and computational efficiency. All models were trained with identical hyperparameters, including a batch size of 32, learning rate of 0.0005, and the AdamW optimizer.

Table 1: Bi-LSTM model configurations for systematic evaluation of BdSL recognition. HS: hidden size, Attn Dim: attention dimension, FC Dim: fully connected layer dimension.

Model	HS	Layers	Attn Dim	FC Dim
Bi-LSTM-1	256	1	128	256
Bi-LSTM-2	128	2	128	256
Bi-LSTM-3	64	1	64	256
Bi-LSTM-4	32	1	32	128
Bi-LSTM-5	16	1	16	32

Computational Resources. All experiments were conducted on the server using NVIDIA 4090 and A6000 GPUs with 128 V-CPUs.

3.4.2 Signer-Independent Evaluation

This study adopts a signer-independent evaluation strategy to ensure robust generalization. Signers *p1–p11* were used for training and *p12–p16* for testing, covering both upper-body (_c) and full-body (_f) views. This guarantees that the model is evaluated on previously unseen individuals, thereby simulating real-world deployment scenarios.

3.5 Baseline Classification

As a prerequisite for our cluster-based analysis, we first established a performance baseline. All five model configurations were trained and evaluated on the full 50-class dataset following a signer-independent protocol. This initial experiment serves two critical purposes: (1) it provides a benchmark against which the performance of our specialized, cluster-based models can be measured, and (2) it generates the confusion matrix that forms the basis for our phonological grouping strategy.

3.6 Phonological Clustering via Confusion Matrix Analysis

To validate our hypothesis that specialized models can better disambiguate similar signs, we first needed to group signs based on their observed confusability. This was achieved through a multi-stage hierarchical clustering process, using the misclassifications from our best-performing baseline model as a proxy for phonological or visual similarity.

Deriving a Similarity Metric from Model Confusion. The foundation for our clustering was the confusion matrix generated by our best-performing baseline model on the full 50-class dataset. The raw confusion matrix of counts, however, is unsuitable for clustering due to class imbalances. There-

fore, we transformed it into a robust distance metric through the following three steps:

1. **Row Normalization:** Each row was divided by its sum, converting counts into recall-like percentages and mitigating the effect of frequent signs.

2. **Symmetrization:** To account for asymmetric confusions, we combined the normalized matrix with its transpose:

$$S = \frac{N + N^{\mathsf{T}}}{2}, \tag{2}$$

   where N is the row-normalized confusion matrix.

3. **Conversion to Distance:** Similarities were transformed into distances for clustering:

$$D = 1 - S, \tag{3}$$

   with diagonals set to zero. Higher similarity thus corresponds to smaller distance.

Hierarchical Agglomerative Clustering. With a valid distance matrix established, we applied Ward's hierarchical agglomerative clustering method (Ward, 1963), which is designed to find compact, spherical clusters by minimizing within-cluster variance. To determine the optimal partitioning of the signs, we generated a dendrogram to visualize the linkage distance at which each merge occurs (see Figure 5). The number of final clusters was determined by selecting a distance cutoff. This was an exploratory process based on visually inspecting the dendrogram to identify a threshold that provided a meaningful balance between intra-cluster similarity (shorter linkage distances) and a manageable number of final groups for our subsequent experiments.

3.7 Cluster-Specific Feature Engineering and Classification

Following the partitioning of the 50 signs into phonologically coherent groups, we proceeded with the core hypothesis of our work: that training specialized models on datasets with custom-engineered features would yield higher recognition performance. This phase involved developing distinct feature engineering pipelines for each cluster based on its composition, followed by a systematic training and evaluation process.

Feature Engineering Strategy. The feature engineering process was tailored to the nature of the signs within each cluster, which were categorized as one-handed, two-handed, or mixed.

One-handed clusters: The feature set was designed to emphasize the dominant (right) hand. Features corresponding to the non-dominant (left) hand were excluded to reduce noise and dimensionality. The engineered feature vector for these clusters included:

1. Anatomically weighted keypoints for the right hand and body, with higher importance given to fingertips and the wrist.

2. Handcrafted relational features such as the Euclidean distance between the thumb and index fingertips (*thumb_index_dist*).

Two-handed clusters: The feature set was designed to capture the complex interplay between both hands. For two-handed signs, we extracted enhanced spatiotemporal features by retaining both left and right hand keypoints along with full body landmarks. To capture meaningful articulations, we applied strategic weighting across joints—emphasizing fingertips, thumbs, and palm knuckles for both dominant and non-dominant hands. Body joints such as shoulders, elbows, and wrists were also weighted to reflect their relevance in upper-body gestures. Beyond raw positions, we computed relative features between the two hands, including inter-hand distance, vertical elevation difference, and fingertip symmetry, all of which were further scaled by predefined weights. Additionally, temporal dynamics were encoded by calculating per-frame velocity for all hand keypoints and relative features.

Mixed Clusters: To avoid bias between one- and two-handed signs, we reverted to the baseline feature set, providing a neutral representation space.

Cluster-Specific Model Training. With the specialized feature sets prepared, we proceeded to the final experimental phase: training and evaluating our clusters, we trained and evaluated each cluster independently with the same five Bi-LSTM model configurations (1 through 5). All training hyperparameters, including learning rate, optimizer, and early stopping criteria, were kept identical to the baseline experiments to ensure a fair and direct comparison. Performance was evaluated using Top-1 Accuracy and Macro F1-score.

4 Experimental Results

To make our dataset more robust and improve generalization, we applied a series of augmentation strategies to the original video samples. Table 2 summarizes the dataset expansion process across all augmentation types. Salt noise was also applied during rotation-based transformations to further increase variability.

Table 2: Dataset expansion with augmentation techniques (50 glosses).

Augmentation Type	Samples	Total
Original	30	1,500
Rotation (20°) + Noise	30	1,500
Left Rotation (40°) + Noise	30	1,500
Right Rotation (40°) + Noise	30	1,500
Zoom-In	30	1,500
Total	**150**	**7,500**

Following data augmentation, we extracted keypoints from all video samples using the MediaPipe Holistic model to capture body and hand motion effectively; only 2D (x, y) coordinates were retained, and facial keypoints were excluded from analysis. Figure 3 illustrates a sample temporal sequence selected via motion analysis.

Figure 3: Selected six(6) annotated frames from the sign *aam*(mango), illustrating distinct temporal phases of the signing gesture. These frames were selected randomly based on motion-aware keypoint displacement.

Input Representation. For the baseline classification experiments, the pre-processing and feature engineering pipeline resulted in a standardized input representation for all 50 signs. Each sign instance was transformed into a matrix of shape 573×150, where 573 represents the uniform sequence length achieved through repetitive padding, and 150 corresponds to the number of keypoint features extracted per frame. This matrix served as the input to each of the five Bi-LSTM model configurations. For the subsequent cluster-based experiments, the sequence length was maintained, while the feature dimension was adapted based on the cluster-specific feature engineering strategy.

4.1 Baseline Performance on the Full Dataset

In the initial phase, all five Bi-LSTM configurations were trained on the full 50-class dataset to establish a baseline, identify the optimal architecture, and generate the confusion matrix for subsequent phonological clustering.

Recognition Performance and Training Cost. The primary performance and cost metrics for each model are summarized in Table 4. The results indicate that Bi-LSTM-1, the configuration with the highest parameter count (1.05M), achieved the best overall recognition performance, with a Top-1 Accuracy of 89.72% and a Macro F1-score of 0.8928. This suggests that for the diverse 50-class problem, a wider model architecture is beneficial. A clear trade-off is visible, as the training time generally decreases with the number of parameters. The most compact model, Bi-LSTM-5, exhibited a noticeable drop in accuracy, indicating that its capacity was likely insufficient to learn the complexities of the full dataset.

Table 3: Recognition performance of Bi-LSTM models on the 50-class dataset.

Model	Accuracy (%)	Macro F1	Params
Bi-LSTM-1	**89.72**	0.8928	1.05M
Bi-LSTM-2	88.80	0.8849	780K
Bi-LSTM-3	89.44	0.8918	380K
Bi-LSTM-4	89.12	0.8891	140K
Bi-LSTM-5	84.80	0.8439	20K

Computational Efficiency Analysis. To address the computational efficiency of our models, we further analyzed their theoretical complexity (GFLOPs) and practical speed (inference time). A lower GFLOPs value indicates a less computationally intensive model, while a shorter inference time is critical for real-world deployment. As expected, a reduction in parameters directly correlated with a decrease in GFLOPs and a substantial improvement in inference speed. For example, Bi-LSTM-4 achieved an accuracy comparable to the larger models but with an inference time of only 0.76 ms, making it almost 12 times faster than Bi-LSTM-1 (8.86 ms). This demonstrates a highly favorable trade-off for resource-constrained applications. We also observed that the training time for the smallest model,

Bi-LSTM-5 (65.98 s), was slightly longer than for the larger Model, Bi-LSTM-4 (57.46 s). This is attributed to slower convergence; the model's limited capacity required more training epochs to reach its peak performance before early stopping was triggered, highlighting that theoretical model size does not always perfectly predict total training duration.

Table 4: Computational efficiency of Bi-LSTM models.[1]

Model	GFLOPs	Train (s)	Infer (ms)
Bi-LSTM-1	0.52	294.83	8.86
Bi-LSTM-2	0.40	174.04	5.18
Bi-LSTM-3	0.17	100.48	2.62
Bi-LSTM-4	0.07	57.46	0.76
Bi-LSTM-5	0.01	65.98	0.69

To further analyze model behavior, we generated a confusion matrix from Bi-LSTM-1 predictions, which served as the foundation for phonological clustering based on inter-class misclassification patterns. For clarity, only a zoomed-in section of the full 50-class confusion matrix(Figure 4) is shown here, highlighting a subset of representative words. The complete confusion matrix is presented in (Figure 7).

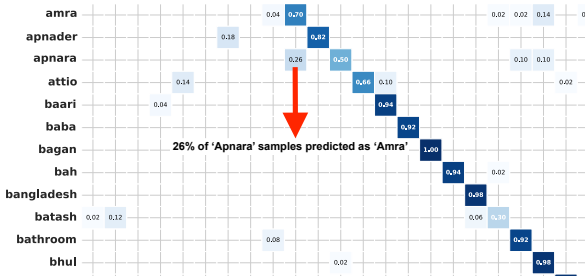

Figure 4: Selected confusion matrix section showing key inter-class confusions for phonological clustering.

The confusion matrix highlights overlaps among visually similar signs. *Amra (we)* was correctly identified in 70% of cases but frequently confused with *bhul (mistake)*. *Apnara (you)* achieved 50% correct recognition, with substantial misclassification (26%) as *amra*,(10%) as bhul. In contrast, *bhul* attained 98% accuracy, showing minimal confusion. These inter-class patterns—particularly between *amra* and *apnara*—are also reflected in the hierarchical relationships shown in Figure 5.

[1]GFLOPs denote the theoretical computational complexity in billions of floating-point operations for a single forward pass. Train (s) indicates the total training time, and Infer (ms) the average inference latency per sample.

The dendrogram (Figure 5) illustrates the hierarchical organization of sign similarity derived from inter-class misclassification patterns. Only a representative portion of the dendrogram is shown here, highlighting selected words. The full one is presented in (Figure 8). The clustering with a threshold of 1.15, chosen empirically to balance granularity and interpretability.

Cluster Analysis

The dendrogram reveals visually and phonologically similar signs, corresponding closely to the confusion trends observed in Figure 4. In particular, signs such as *amra (we)*, *apnara (you)*, and *bhul (mistake)* frequently co-occur in the same cluster due to overlapping hand trajectories, as seen in both the confusion matrix and the dendrogram. Similarly, *batash (wind)* and *bon (forest)* exhibit consistent misclassification. These findings validate the use of confusion-driven clustering to group signs sharing structural and articulatory similarities.

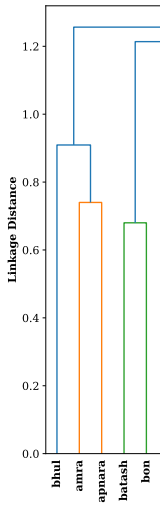

Figure 5: Selected section of the hierarchical dendrogram.

The cluster composition is summarized in Table 5.

Table 5: Clusters of visually or phonologically similar signs derived from confusion-based similarity.

ID	Words in Cluster
1	amra, apnara, bhul
2	batash, bon
3	chocolate, golarhaar
4	chikitsha, dengue
5	chih, cricket
6	aam, aat, abhawa, ador, agun, akash, amader, amar, ami, apnader, attio, baari, baba, bagan, bah, bangladesh, bathroom, bichana, boi, boka, boro, brihospotibar, budhbar, chabi, chacha, chachi, chagol, chalak, chata, choto, doctor, dorja, durbol, football, glass, goru, hata, hospital, janala

As the clusters were derived from confusion-driven phonological similarities, we analyzed hand usage within each cluster to guide targeted feature engineering. Feature design emphasized dominant-hand dynamics for one-handed clusters and bilat-

eral coordination for two-handed ones. For the mixed cluster (Cluster 6), no specialized feature adaptation was applied due to its heterogeneity. After obtaining the clusters from the dendrogram, we applied tailored feature engineering and classification techniques for each cluster, as briefly outlined in the methodology. Following the tailored feature extraction, we trained Bi-LSTM models independently for each of the six clusters using the same five architectural settings as in the baseline to maintain experimental consistency. The resulting classification performance—measured by Top-1 accuracy and F1-score on a signer-independent test set—is summarized in Table 6.

Table 6: Cluster-wise Top-1 Accuracy (%) Across Five Bi-LSTM Settings

Cluster	Bi-LSTM-1	Bi-LSTM-2	Bi-LSTM-3	Bi-LSTM-4	Bi-LSTM-5
Cluster 1	**84.67**	**84.67**	81.33	83.33	77.33
Cluster 2	**91.00**	**91.00**	88.00	90.00	84.00
Cluster 3	97.00	99.00	**100.0**	99.00	95.00
Cluster 4	**90.00**	**90.00**	**90.00**	**90.00**	88.00
Cluster 5	95.00	93.00	85.00	88.00	**99.00**
Cluster 6	**94.00**	92.82	93.38	91.59	87.59
Weighted Avg	93.32	92.40	92.24	91.12	87.60

This evaluation allows us to examine how model complexity interacts with cluster composition and whether confusion-aware feature design leads to measurable improvements. Cluster-level evaluation (Table 6) shows that recognition accuracy improves across nearly all groups, with Clusters 2–6 consistently surpassing the baseline. Weighted averages confirm these gains: Bi-LSTM-1 improves from 89.72% to 93.32%, Bi-LSTM-2 improves from 88.80% to 92.40%, Bi-LSTM-3 rises from 89.44% to 92.24%, and Bi-LSTM-4 from 89.12% to 91.12%. Even the lightest model, Bi-LSTM-5, recovers from 84.80% to 87.60%. These results demonstrate that clustering consistently enhances recognition, regardless of model scale. Most importantly, clustering redefines the accuracy–efficiency frontier. Table 7 compares the strongest baseline (Bi-LSTM-1) with the clustered results of Bi-LSTM-2, Bi-LSTM-3, and Bi-LSTM-4. Each clustered configuration not only surpasses the baseline accuracy but does so with dramatically fewer parameters. For instance, Bi-LSTM-2 achieves 92.40% accuracy using only 780K parameters (25.7% of B1), while Bi-LSTM-3 maintains 92.24% with just 380K parameters (36.2% of B1). Bi-LSTM-4, the most efficient model, reaches 91.12% accuracy with only 140K parameters, representing a reduction of 86.7% in parameter count compared to B1.

Table 7: Comparison of Baseline and Clustered model efficiency and performance. Abbreviations: B1 = Bi-LSTM-1, B2 = Bi-LSTM-2, B3 = Bi-LSTM-3, B4 = Bi-LSTM-4.

Type	Configuration	Accuracy (%)	Total Params
Baseline	B1	89.72	1.05M
Clustered	B2	**92.40**	780K
Clustered	B3	**92.24**	380K
Clustered	B4	**91.12**	**140K**
	B2 uses 74.3% of B1 parameters (↓**25.7%**)		
	B3 uses 36.2% of B1 parameters (↓**63.8%**)		
	B4 uses 13.3% of B1 parameters (↓**86.7%**)		

Table 7 presents a direct comparison between the strongest baseline model (Bi-LSTM-1) and the efficient clustered configurations (Bi-LSTM-2, Bi-LSTM-3, and Bi-LSTM-4). Results show that clustering improves accuracy across all models while significantly reducing parameters, enabling lightweight Bi-LSTMs to achieve competitive performance suitable for resource-constrained settings.

5 Conclusion

This study presented a hierarchical framework for BdSL recognition, combining confusion-matrix-based clustering with phonological feature engineering and Bi-LSTM sequence modeling. Cluster-level evaluation showed that Clusters 1–5 consistently outperformed baseline recognition, with weighted average accuracy improving from 89.72% to 93.32% for the largest model, and even compact models achieving up to 92.12% with fewer parameters. These results demonstrate that clustering reduces inter-class ambiguity and enables lightweight models to achieve competitive performance in large-vocabulary recognition tasks, while also lowering training time and inference cost. While a Bi-LSTM was used for its balance of performance and interpretability, the framework is model-agnostic and can be extended to more advanced architectures in future work.

6 Limitations

While the results are promising, several limitations persist. A key challenge arises from the high visual similarity of signs within Clusters, where many signs exhibit nearly identical hand trajectories and articulations. This resemblance complicates differentiation, particularly in signer-independent scenarios where variations in articulation style and execution speed further obscure distinctions. These findings underscore the inherent difficulty in distinguishing signs with substantial inter-class similarity using conventional keypoint-based methods. The temporal and spatial similarities between these signs often confuse the classifier, especially under signer-independent evaluation. Figure 6 presents representative frames for these two signs from Cluster 2, demonstrating the difficulty in distinguishing between them based solely on visual features.

Figure 6: Representative frames from signs in Cluster 2. Each sign is displayed with four frames arranged in two rows to highlight handshape, orientation, and movement. Visual similarity among signs contributes to frequent misclassification.

Second, the keypoint extraction process relied entirely on MediaPipe Holistic, which occasionally failed in frames with motion blur, occlusion, or fast signing. Incomplete or inaccurate hand landmark detection led to the exclusion of such frames, which may have affected the recognition of signs that require fine-grained temporal information. While preprocessing helped mitigate noise, a more robust multi-modal or interpolation-based method could further improve reliability.

Lastly, the experiments were conducted on a 50-class subset of the SignBD-Word dataset. While this subset allowed for controlled experimentation, it does not fully represent the complexity of large-scale real-world BdSL recognition tasks. Future evaluations on the full 200-class dataset and other benchmark sign language datasets are necessary to assess scalability and generalizability.

References

Aonmoy Das, Ananna Dev Aishi, Masbah Uddin Toha, and Md Fazlul Kader. 2025. Bangla sign language translator for deaf and speech impaired people using deep lstm. *Multimedia Tools and Applications*.

Manuel Gil-Martín, María Villa-Monedero, Andrzej Pomirski, Daniel Sáez-Trigueros, and Rubén San-Segundo. 2023. Sign language motion generation from sign characteristics. *Sensors*, 23(23).

Kazi Reyazul Hasan and Muhammad Abdullah Adnan. 2025. Empath: Mediapipe-aided ensemble learning with attention-based transformers for accurate recognition of bangla word-level sign language. In *Proceedings of the International Conference on Pattern Recognition*, pages 355–371, Cham. Springer Nature Switzerland.

Md Shamimul Islam, A. J. M. Akhtarujjaman Joha, Md Nur Hossain, Sohaib Abdullah, Ibrahim Elwarfalli, Md Mahedi Hasan, and Muhammad Imran Khan. 2024. Multi-view bangla sign language recognition: A new word-level video dataset. In *Proceedings of the 2024 International Conference on Activity and Behavior Computing (ABC)*, pages 1–6.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. 2019. Mediapipe: A framework for perceiving and processing reality. In *Proceedings of the Third Workshop on Computer Vision for AR/VR at IEEE CVPR*.

S. M. Mostavi Mashkur Mahin, Md. Raiyanul Islam, and Sk. Md. Masudul Ahsan. 2023. Phrase level bangla sign language recognition using keypoints from hand gesture video. In *Proceedings of the 2023 International Conference on Next-Generation Computing, IoT and Machine Learning (NCIM)*, pages 1–6.

Donghoon Oh, Jeong-Sik Park, Ji-Hwan Kim, and Gil-Jin Jang. 2021. Hierarchical phoneme classification for improved speech recognition. *Applied Sciences*, 11(1):428.

Wasupon Phothiwetchakun and Thanawin Rakthanmanon. 2021. Thai fingerspelling recognition using hand landmark clustering. In *Proceedings of the 2021 25th International Computer Science and Engineering Conference (ICSEC)*, pages 256–261.

Husne Ara Rubaiyeat, Ahsan Habib, Md. Kamrul Hasan, and Hasan Mahmud. 2025. Bdslw60: A word-level bangla sign language dataset. *Multimedia Tools and Applications*.

Ataher Sams, Ahsan Habib Akash, and S. M. Mahbubur Rahman. 2023. Signbd-word: Video-based bangla word-level sign language and pose translation. In *Proceedings of the 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pages 1–7.

P. Schneider, R. Memmesheimer, I. Kramer, and D. Paulus. 2019. Gesture recognition in rgb videos using human body keypoints and dynamic time warping. In *Computer Vision – ACCV 2018 Workshops*, pages 311–326. Springer.

Mirza Md Shakil, Al Amin, Tahmid Anam, Molla Rashied Hussein, Tahira Alam, and Jamim Hossain Chowdhury. 2024. Daily life conversation in bangla sign language using machine learning technique. In *Proceedings of the 2024 IEEE 6th Eurasia Conference on Biomedical Engineering, Healthcare and Sustainability (ECBIOS)*, pages 288–292.

OpenCV Team. 2024. Open source computer vision library. Version 4.8.0, `https://opencv.org/`.

Joe H. Ward. 1963. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301):236–244.

A Supplementary Figures

The following figures present full-width versions of the confusion matrix and dendrogram for detailed inspection, as referenced in the main text.

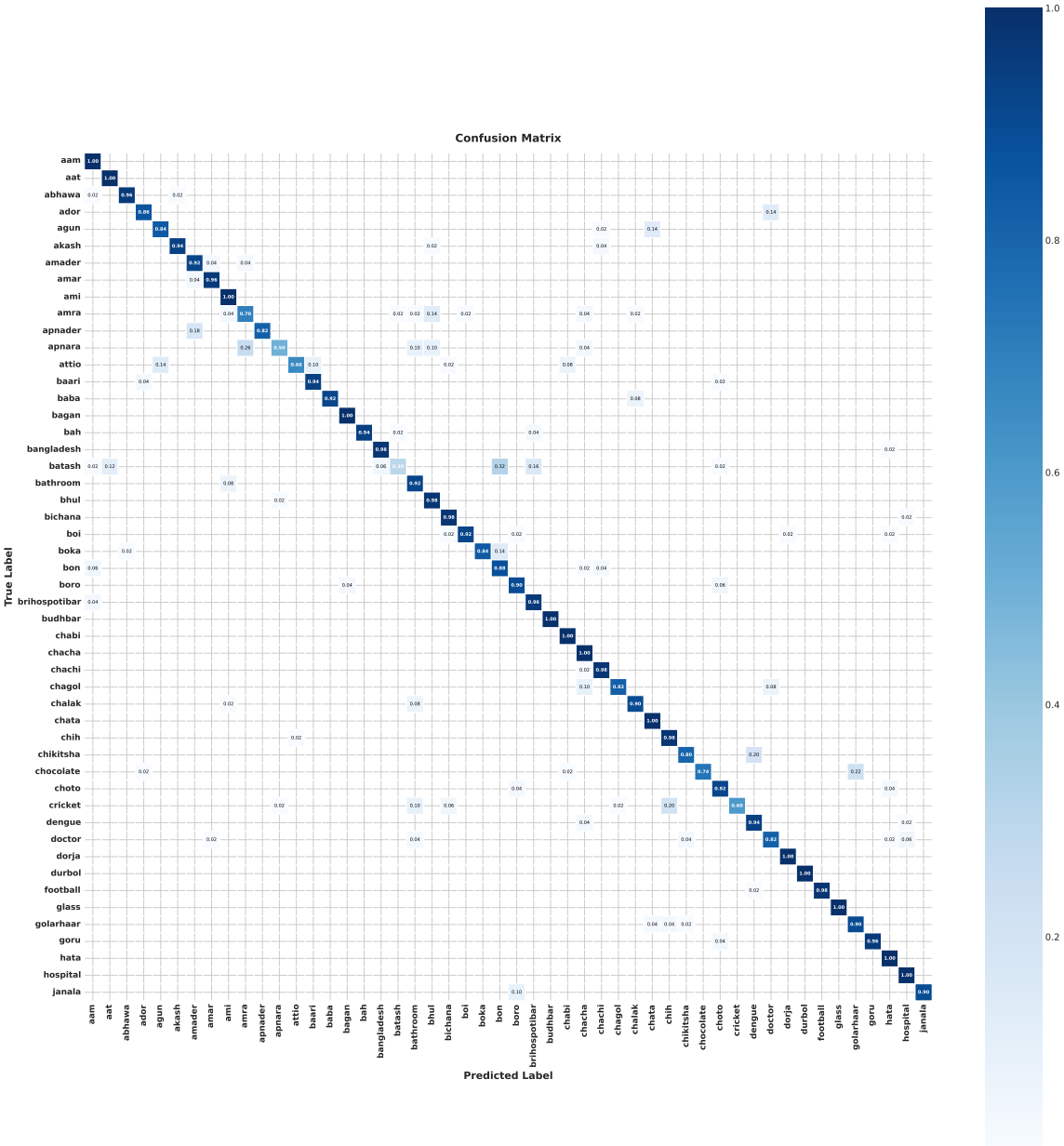

Figure 7: Complete 50-class confusion matrix from the best-performing baseline model (Bi-LSTM-1). The strong diagonal indicates high overall accuracy, while off-diagonal cells reveal systematic confusions between phonologically similar signs, which justifies the subsequent clustering analysis.

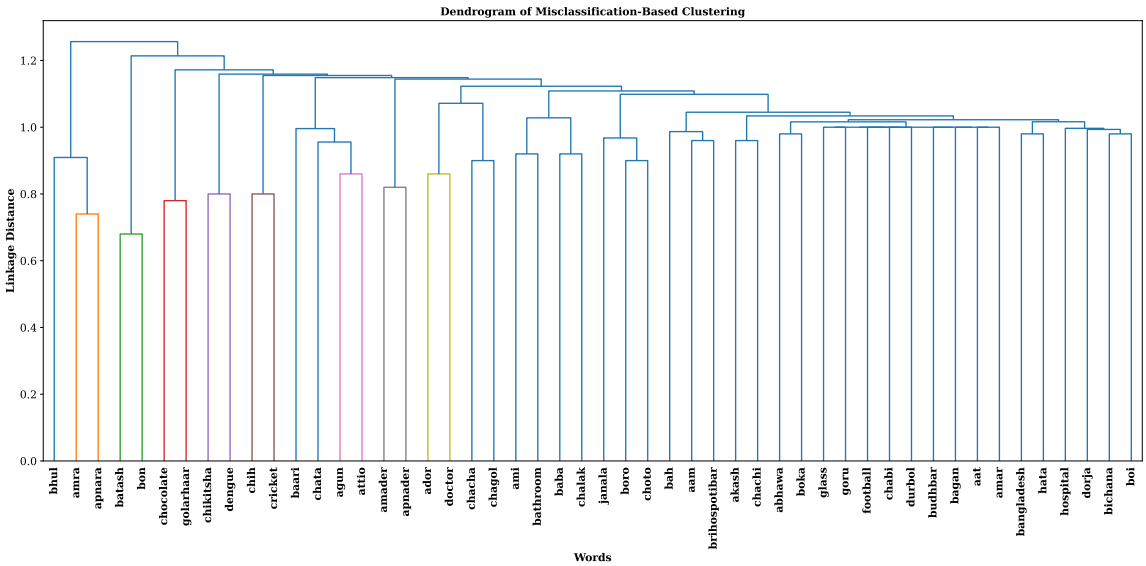

Figure 8: Full hierarchical dendrogram of all 50 signs, generated using Ward's linkage on the distance matrix derived from model confusions.