TWMS J. App. and Eng. Math. V.15, N.9, 2025, pp. 2331-2351

EXTENSIVE ERROR DERIVATIVE REVIEW OF LSTM MODELS WITH SIGN LANGUAGE INTERPRETATION

HARAPRIYA KAR[1], VISWANATHAN P.[1]*, §

Abstract. LSTM models are essential for systems that translate sign language, where the model suffers from error loss when processing data. LSTMs reduce error propagation by continuously calculating gradients, unlike traditional back propagation, which causes exponential error accumulation. This paper investigates error flow in bidirectional, hierarchical, and probabilistic long short-term memory models (LSTMs). While hierarchical LSTMs employ multitask learning to anticipate inputs and outputs, minimizing compounding mistakes reliably, bidirectional LSTMs reduce truncation errors. Model accuracy is increased by optimizing the gradients and parameters. This research offers a thorough evaluation of LSTM models from 2021 to 2024, examining their effectiveness in sign language recognition systems by analyzing both accuracy and loss.

Keywords: RNN, LSTM, Bidirectional LSTM, Bayesian LSTM, Hierarchical LSTM, Parametric.

AMS Subject Classification: 83-02, 99A00

1. Introduction

Sign language recognition systems are multi-dimensional data processing which require multi-layer networks to optimize the recognition system. The Deep learning models based on recurrent neural networks will have a high effect on the sign language interpretation systems [1]. The error analysis of the network model plays a major role in optimizing the accuracy. The backpropagation algorithm is one of the most used neural learning techniques that learn weight based on gradient descent [2, 3] in multi-layer [4] networks of sign recognition systems. It trains the data backward from the output layer to the hidden layer using the activation function in which neurons are differentiable. The temporal evolution of the backpropagation error signals flows backwards in time. Recurrent learning in training the sign language data occurs exponentially due to rapid increases in the magnitude of the weights which will extend the evaluation of the error rate raises the training time and complexity.

[1] Computer Science and Engineering, Vellore Institute of Technology, Vellore, Tamilnadu 632014, India.
e-mail: harapriya1999sa@gmail.com; https://orcid.org/0009-0001-1526-2754.
e-mail: pviswanathan@vit.ac.in; https://orcid.org/0000-0002-9337-8760.
* corresponding author.
§ Manuscript received: August 02, 2024; accepted: October 23, 2024.
TWMS Journal of Applied and Engineering Mathematics, Vol.15, No.9;

Typical Backpropagation Recurrent Neural Network (BPRNN) [5] process the sign data point in a sequential direction emerged at a high level. It limits the feeding of signals in temporal [6] periods longer than the massive space in dynamic classification [7]. The error backpropagation issues and fed-back signal limitations are extended by increasing the steps based on the network size using Long short-term memory (LSTM). LSTM [8, 30] with gradient model learned to bridge temporal periods longer than the noisy and incompressible input sequences without limiting the short time lag. It enforces a continuous flow of errors in the internal states of special units, explodes and vanishes as long as the gradient process at specific points is terminated which leads to the issue of truncation error.

The bidirectional LSTM (BiLSTM) [10] architecture is the didactic approach which raises the storage and retrieval concerns of sign data to avoid the one-step truncation by the implementation of a total error gradient in the network. It is further optimized by the phenomena of training the input by propagating forward and backward in two different LSTM networks connected to the same output layer. However, this architecture fails due to the segmentation of temporal data which needs probabilistic [11, 12] backward recurrence to maintain the variance. The Bayesian LSTM(BaLSTM) reduces the requirement of backward recurrence by processing the input sequence with activation based on the context with the help of a probabilistic gate to optimize [13] less training of sign data. The BaLSTM model evaluates the dynamic data effectively but for multidimensional [14] aspects of time series [15, 16] require interrelated dynamical data for sign data. The hierarchical model [17] approach evaluates sign data in time series [18] analysis in the aspect of multilevel for predicting future input and past output. It can be analyzed coherently [19] with a co-attention [20] prediction mechanism to decide the future output dynamically. The hierarchical LSTM [21, 22] processes the high-level and low-level context by connecting two LSTMs at each time step related to multilevel layers and deliberatively refining sequential input.

Future Intention Estimation (FIE) with sign interpretation improves the forecasting of viewport transition constructed to capture the temporal [23] correlations between past, present, and future output estimated using Hierarchical BaLSTM (HBaLSTM) [12]. It is devised with additional modeling of inter-subject uncertainty in a training system named Hierarchical Bayesian inference (HBI) using hierarchical joint Gaussian distribution. It approximates the posterior distribution across network weights, anticipates viewport change on specific output by combining expected future information with identical temporal dependency. The parameters and computation resources of the conventional units in recurrent cells are expensive in deep recurrent nets.

The stackable recurrent cell [24] can reduce the parameters by evaluating the weight factor < 1, leads to zero mean symmetric distribution of sign data. It shrinks the overall gradient and multiplicatively affects the Jacobean parameter [8], avoids output gates to carry the similar information. It reduces the parameters of the stackable Recurrent (STAR) unit of the input gate, and output gate which merges with the forget gate [25, 26]. It further preserves the reduction of the gradient [3] magnitude of deep RNN [27] lattice, with fewer parametric updates that induce the non-linearity to saturate; large gradients can sometimes represent the prelude to vanishing gradients, reducing the computation resources.

The literature review deals with various LSTM models concerning error flow with the directional, time series, probabilistic, and multi-layer with parametric principles in sign language interpretation system. Our extensive review of the LSTM models derives the emerging challenges faced in sign data in the context of error signals and are reported

theoretically and experimentally in our paper. LSTM models are analyzed in various aspects of computing consistent gradient with the error flow analyses using loss and accuracy function to define the efficiency of the model. The main objectives of the review is

(1) To scrutinise the gradient flow error analysis of the loss function for the various LSTM models.
(2) To derive the complexity of the error flow analysis from the previous LSTM models to current LSTM model in the aspects of single level to multilevel.
(3) To analyze the impact of loss functions with the prediction accuracy of LSTM models using sign language interpretation system with 2D to 3D dataset.

The Section 1 explains the introduction of the various LSTM models. The design of the directional and the probabilistic[11] LSTM with reduction of error back propagation issues and the limitations of the error signals with future prediction is derived in Section 2. Section 3 derives the evaluation of multidimensional data in interrelated dynamics within the time series context using a multilevel LSTM approach with reduction of computational resources by limiting the parameters. Section 4 discusses the comparative analysis of sign language using multiple LSTM models. In Section 5 derives an accurate experimental evaluation of different LSTM models for sign language is presented in terms of accuracy and loss metrics. Ultimately, Section 6 delivers a summary of the outcomes drawn from the study of the survey in order to determine the issues with the LSTM model. These issues include the incompatibility of the dataset's size and shape, the inability to comprehend the error flow, and the processing of 3-dimensional video and signs while interacting with real-world situations.

2. Single Level RNN and LSTM

The BPRNN architecture [24] integrated with a robust gradient-based learning algorithm [2] helps learn to bridge temporal encompasses longer than a thousand times without sacrificing its short-time lag capabilities. The BPRNN [28] improves the sensitivity in output depend on the area of network. The objective function minimizes overall area of network within timestamp (T) $t' \leq T \leq t$of the non input unit (N) of the network unit (I, J) by evaluating the distance metric between prior weighted output d_N and the target y_N.

$$E(T) = \frac{1}{2} \sum_{N \in I} [d_N(T) - y_N(T)]^2 \tag{1}$$

The total error E_A in equation (1) is the sum of the error rate of the epochs which is called gradient descent error equation (2) evaluated based on the gradient error weight update [29].

$$\nabla_w \; E_A \left(t', t+1\right) = \nabla_w E_A \left(t', t\right) + \; \nabla_w E \left(t+1\right) \tag{2}$$

The sensitivity of the output is determined by p_N equation (3) and avoids the back propagation [29] with respect to time (T), for the new data. Where $W_{[I,J]}$ defines the weight of the gradient descent equation (4).

$$p_N = \frac{\partial y_N(T)}{\partial W_{[I,J]}} \tag{3}$$

$$\begin{aligned} \Delta W_{[I,J]}(T) &= -\eta \sum_{k \in I} \frac{\partial E(T)}{\partial y_k(T)} p_N \\ &= -\eta \sum_{k \in I} E_A \times p_N \end{aligned} \tag{4}$$

The weight is updated at the time(T) makes the backpropagation error signal equation (5) of unit cell I_i for the input X.

$$Z_{N,I_i}(t+1) =) = \sum_{l} W_{[(N,I_i),l]} X_{[(N,I_i),l]}(t+1))$$

$$\text{with } \mathrm{l} \in pre(N) \tag{5}$$

$$= \sum_{J_i \in I} W_{[NJ_i]} y_J i(t)) + \sum_{i \in I} W_{[N,i]} y_i(t+1)$$

$$J_{I_i}(T) = f`_{I_i}(z_{I_i}(T) \sum_{J \in I} W_{[I,J]}(T+1)) \tag{6}$$

The (I_i) propagated fully in the time slack factor ranges $(T - T_i) > 1$ in between the output layer neuron and the arbitrary neuron J_i (6) causes the exponential flow of error. The vanishing error in the time lag affects the weight resulted in dynamic error flow on RTRL. The LSTM bridges [30] the minimum time lag, computes discrete time steps which reduced the time steps based on preserved error constant error carousels (CEC) [9] handled by multiplicative gates. To ensure the constant error flow, derivative of (7) is initialized with 1 defines f_I linear. It uses the special unit cell in $y_I(8)$ makes constant error flow directly accesses the network handled by multiplicative gate units. It is emerged itself in the single connection directly proportional to the error flow of next time step (t+1). Hence the error flow y_I becomes linear, derives the activation function z_I acting as constant error flow with the weight W that ranges to 1.0, since the weighted magnitude is not affected and it enhances the storage of LSTM over the maximum period of time steps.

$$f_I(z_I(T)) = \frac{z_I(T)}{W_{[I,J]}} \tag{7}$$

Hence the function (f) becomes linear and it activated over the constant time stamp(T)

$$Y_I(T+1) = f_I(z_I(T+1) \tag{8}$$

$$f_I(y_I(T)W[I,I] \tag{9}$$

The CEC backflow remains constant even though there is a need of additional weight input and output. It makes the confliction in weight update when it stores and ignores the input is having the same weight. In order to handle conflicting weight updates, the LSTM optimizes [31] the CEC by updating input and output gates which connects extra memory cells with the network layers. The activation function ranged from [0, 1] for the sigmoid threshold units Y_I is initialized in the input Y_{In} (10) and output Y_O (11) control the signals in the input and output gates, the gate closes when activation is nearly zero. Gates can also learn to protect the data kept in memory cell from interruption due to unrelated impulses. It solves the confliction of weight but leads to the exploding of vanishing error [32] which can be resolved by the initialization of the forget gate [25, 26]. The signal is scaled from network to memory cell effectively in the activation function of the forget gate Y_f (12), states to 0 or 1.0 preserves the memory cell over time whereas 0 removes the memory cell such that it acts like RNN. It increases the complexity of an LSTM unit, also referred to as a block of memory.

$$Y_{In}(T+1) = f(\sigma(W_{[IJ]} * [J_{[it]} - z_{[J-1]}] + Y_{[In]}(T))) \tag{10}$$

$$Y_O(T+1) = f(\sigma(W_{[IJ]} * [J_{[it]} - z_{[J-1]}] + Y_{[O]}(T))) \tag{11}$$

$$Y_f(T+1) = f(\sigma(W_{[IJ]} * J_{[it]} - z_{[J-1]} + Y_{[f]}(T))) \tag{12}$$

By learning how to regulate access to the content of memory cells, the output gates are capable of protecting neighbouring memory cells from disruptions enriching from the

unit cell I. In order to regulate the access, the first layer of the LSTM gate is started sequentially by the input, output, and forget gates. This hidden state is referred to as short-term memory Y_H (14) if the output gate and the cell state initiated in the hidden state which turns out to long term memory cell state Y_C (13) and generate the current output. An activation function called tanh exists in the cell state, and it ranges from -1 to 1.

$$Y_C(T+1) = Y_f(T) \times Y_C(T) + Y_{In}(T) \times N_t \tag{13}$$

$$Y_H(T+1) = Y_O(T) \times tanh(Y_C) \tag{14}$$

$$Y_N \ = tanh(w_N \times [I_t - z_{I-1}] \ + f_N) \tag{15}$$

The forget and input gates employ a sigmoid activation function that oscillates between 0 and 1, allowing the cell state to be added and subtracted from the memory cell. This memory cell serves as a long-term memory for making further predictions. The cell state Y_C (13) is the memory cell functions as a new state Y_N (15) which follows the stacking function. Thus, it is evident that the primary purpose of multiplicative gated units' main function is to allow or prohibit continuous error flow via the CEC. If the unit u is similar to the output gate of the memory block stops the propagation of error due to truncation. It leads to restrict the updating of weight during backward propagation making inconsideration of recurrent connection. The multiplicative gates are the issues that it may permit or prohibit the continuous error flow via CEC in dynamic Environment [7, 33]. The Bidirectional LSTM[10] overcomes the limitation of time series data over the input and output vectors. Bidirectional LSTM splits the neuron into the input vector X_i(16) in forward time direction.

$$X_i(T) = x1, x2, x3, ...x(t-1), x(t) \tag{16}$$

The output vector (Y_i) (17)in backward direction

$$Y_i(T) = y1, y2, y3, ...y(t-1), y(t) \tag{17}$$

The projected outputs from the forward pass feed the BRU over all of the input data for a single time stamp (t) from t=1 to T and backward from T to 1. In backward pass, it feeds performing the output neurons forward pass from t=T to 1 and backward states to the signal from 1 to T. When two LSTM networks connected to the same output layer are enhanced by the process of forward and backward passes, the whole error gradient calculation is implemented throughout the recursive time stamp (t). With both time directions maintain the past input information and current future time frame minimizes the objective function without delay. As the forward and backward signal recursively flows over, the time stamp (t) removes one step truncation leading to the occurrence of unimodal regression and estimation of the conditional probability for all the available input data, in a full series of classes for the duration time (t). The outputs derived are statistically dependent which makes difficulties in the evaluation of the temporal data. The Bayesian LSTM [12, 34] avoids the limitation of the segmentation of temporal output data over the two-pass algorithm (forward pass and backward pass) with the aspect of future predictions. The feature's presence or absence throughout the entire input sequence is processed by the BRU but still, the LSTM shows that it is necessary to assume that the entire input sequence contains the feature, or it does not contain the same feature [35]. It permits various responses from an activation indecency of context-specific inputs based on the conditional probability leading to probabilistic input gate (18) for the temporal observations.

$$p(C_t|X_{t,t-1}) \propto p(X_t|C_t)p(C_t|X_{t-1}) \tag{18}$$

The H_t is the observed sequence of the event for the feature [36] Φ_t with conditional prior probability $p(\Phi_t|X_t)$ of the feature ft up to the time stamp (t). The inverse of the observed sequence $(1 - H_t) = p(\Phi_t|X_t)$ is not time dependent. Hence it derives the independent of probability for the given parameter θ. The joint probability with θ (19) condition is as follows.

$$H_t = p(\Phi_t|(\theta, X_t) \tag{19}$$

To improve the correctness, condition θ is dropped resulting in bi which are sigmoid function and multiplicative feedback (20) derived as follows.

$$\begin{aligned} \frac{P(\bar{\Phi}_t|X_{t-1})}{P(\Phi_t|X_{t-1})} &= \frac{1 - H_{t-1}}{H_{t-1}} \\ &= \log(H_{t-1}) - \log(1 - H_{t-1}) \end{aligned} \tag{20}$$

Probabilistic forget and input The dependency of the context with the prior context throughout the storage unit is improved by the forget probabilistic approach. The indicative event characteristics of the cell state Y_C over the ϕ_i is initialized with 0 or 1 based on the relevancy and irrelevancy of the context. The storage unit of $z_t = P(Y_C = 1|X_t)$ and inverse of $1-z_t = P(Y_C = 0|X_t)$ foreseeable by the network is based on the probability assigned. The relevant context with the cell state=1 which is by default independent but due to prior probabilistic constant derived from ϕ_i makes it dependent on the context in time stamp t and t-1 prediction of the network.

The layer-wise and unit recurrence of the probabilistic forget gate derives the conventional linear aggregation of prior outcomes and a prior probability P with the input gate $P(Y_I)$ and follows with hidden gate $P(Y_H)$. It derives the output z_t for the input and hidden unit where the prior probability P_i is applied with the logistic regression $F(z_t)$ recursively concerning the current and previous terms of Yht with weighted input. It simplifies the process of memory cell state Y_C with the context if it is relevant then the feature which is retained and flows towards recurrence can lead to predict the dynamic data.

$$\begin{aligned} F(z_t) &= logit([1 - (z_{t-1})]P + z_{t-1}(Y_H t - 1, I) \\ &= logit(z_{t-1}(W_t Y_H t - 1 + c - P) + P) \\ &= logit(z_{t-1}(W_t Y_H t - 1 + c - P) + P) + \beta \end{aligned} \tag{21}$$

The single recurrence is applied with unit-wise and layer-wise probabilistic approach equation (22) to make the network robust and the adhoc gate is retained back. The linear function with the recursive approach with the hidden probabilistic gate works with the context of 0 and 1, the approximation of the log h function estimated for the unknown propagation of h with the prior and constant β (23) The weight factor W is initialized with various gates of LSTM and it is not normalized in the forward pass using bias vector b which emerges backward pass with the probabilistic approach derive the hidden and new state gate as follows

$$PY_{Ht} = \sigma(W_H Y_{In} + b_H + z_{t-1}) \odot (W_H Y_{Ht-1} + b_r) \tag{22}$$

$$PY_{Nt} = \tanh(W_N Y_{In} + b_{In} + z_{t-1}) \odot (W_{HN} Y_{Ht-1} + b_H) \tag{23}$$

The probabilistic forget gate derivations aim at formalizing the LSTM's CEC, resulting in the resemblance of a GRU's reset gate(r). It is determined and upgraded of the gate in the Gated recurrent unit. On considering a binary state variable, r with a value of 1 implies that there is no relevancy in current input, and a value of 0 means irrelevancy. Such that the probability $r_t = P(p_t = 1|X_t)$ and the inverse is $(1 - r_t) = P(p_t = 0|X_t)$ (24) in which there is no relevancy in current input, then ϕ_t it is depending on ϕ_{t-1} a

specific event feature ϕ_i to estimate the probability of the input prediction using equation (26) and probability of the hidden is estimated by using equation (15).

$$r_t = PY_{In}P(\phi_t Y_{In}|X_t) + (1 - r_t, Y_{In})Y_{Ht-1}, Y_{In} \tag{24}$$

$$F(PY_{Ht}) = ((1 - z_t) \odot Y_{Nt} + z_t Y_{Ht-1}) \tag{25}$$

$$F(PY_{In}) = \sigma(WzY_Int + b_H + W_H Y_{Ht-1} + b_H)) \tag{26}$$

Using a prior probabilistic method, the single-level LSTM model based on Bayesian probability efficiently assesses the dynamical data. Prediction is greatly aided by the assessing of the interconnected dynamics when dealing with multidimensional data flows. The use of a single-level LSTM model in the asses of multi-dimensional dynamics leads to an inability to perform optimal prediction in the time stamp (t) due to the compounding of error.

3. Multilevel LSTM Models

Hierarchical LSTM [22] combines low-level and high-level fusion of time series [18] analysis and employs multitask learning which can avoid the compounding of error. Hierarchical LSTM [37] memory concurrently works with the subsequent interrelated dynamics over the time stamp(t) and output of the sub-memory cell ($Y_C t$), which emerged with the concurrent LSTM. The concurrent LSTM fused with the new cell state gate ($Y_N t$), sub-storage unit co-memory cells. Rather than using single-level coherent motion, HLSTM selectively integrates and stores the information in synchronous LSTM units using multiple sub-memory units. The single sub-memory hidden state ($Y_H t$) is determined by the input (Y_T) for the time stamp T of various levels of inputs. Thus, evaluating of the single dynamic yields the single inter-related dynamics at each time stamp (t). $Y_1 \in R|T = 1, Y_2 \in R|T = 2....Y_p \in R|T = 1, 2...t$ where p is the interacting unit For the inputs $Y_H^s|T = 1, 2, ...t$ Each sub memory unit (s) initiated with input gates, forget gates, and sub-memory cells at the time stamp (t). Cell gates transfer interrelated motion memories from sub-memory units to a new co-memory cell defined in all the hidden states. The co-memory cell selectively integrates memories and interrelated information treated as single-level coherence related with interrelated dynamics over the time stamp(t) by stacked co-LSTM units. It coherently added with the input gate$Y_I n^s$ using equation (27) and the forget gate ${Y_F}^s$ using equation (28) with the additional integration of input modulation gate $(Y_N)^s$ using equation (29) and sub memory cell gate $(Y_C)^s$ using equation (30) over the time stamp(t) derives the interrelated dynamics it does not lead to analyze the future intention of the multidimensional data.

$$Y_{In}^s = \sigma(WY_{In}^s \cdot Y_H^s + WY_{In}Y_H^s \cdot Y_{Ht-1} + b_{YIn}^s) \tag{27}$$

$$Y_F^s = \sigma(WY_F^s \cdot Y_H^s + WY_F Y_H^s \cdot Y_H t - 1 + b_{YF}^s) \tag{28}$$

$$Y_N^s = \phi(WY_N^s \cdot Y_H + WY_N^s \cdot Y_{Ht-1} + b_{YN}^s) \tag{29}$$

$$Y_C^s = Y_F^s \odot Y_{Ct-1}^s + Y_I nt^s \odot Y_N t^s \tag{30}$$

Where s=1,2...p The temporal co-relation between the past, present, and future evaluations are abstained in the Hierarchical LSTM model leading to performance loss of the predicted outcome in the optimal time stamp(t). The future intention estimation is based on two factors viewport transition and subject-specific variation. The Hierarchical Bayesian LSTM model derives the above factors to forecast the previous output and future input. It leads to a reduction of error in time series analysis which is acquired from Bayesian fully connected layer (FC) layers. The process of transition in the temporal domain provides variations between the parameters of hierarchical Bayesian LSTM. It evaluates a cell with unique Hierarchical Bayesian Inference (HBI) trained to learn interrelated subject uncertainty. The HBI is derived by evaluating the minimum posterior

distribution with relative entropy divergence (D_{rn}) using equation (31) used to analyze the optimal variation distribution for the approximation of interactable posterior distribution.

$$\min_{\mu_\gamma,\sigma_\gamma,\mu_\eta} D_{rn} o_J(W_L,\gamma_L,\eta)|||DP(W_L,\gamma_L,\eta\|D) \tag{31}$$

Where $o_J(WL,\gamma_L,\eta)$ is considered as the variational approximation to $DP(WL,\gamma_L,\eta|D)$, and $\mu_\gamma,\sigma_\gamma,\mu_\eta$ are the corresponding parameters to be learned. The variable approximation is typically structured in a hierarchy of 3,1 and 2 as $o_J(W_L,\gamma_L,\eta)$, $o_J\gamma_L$, and $o_J\eta$ HBI is executed arbitrarily to process the multidimensional data in the particular period (t) using the weights derived from the inferred Gaussian distributions to evaluate the interrelated subject uncertainty. The hierarchical distribution o_J is modeled by Gaussian with average μ_γ and variance of prior σ_γ^2 using equation (32).

$$o_J^\gamma = N(\gamma\|\mu\gamma,\sigma_\gamma) \tag{32}$$

The inter subject variance does not require γ instead that η using equation (33) with W_L using equation (34) can be used to formulate Gaussian posterior distribution to solve the inter-subject variance for the prediction of the multi-dimensional data.

$$o_J(\eta) = N(\eta\|\eta,\sigma_0^2 I) \tag{33}$$

The mean of the data μ_η is defined as the parameter and variance of the data σ_0^2 is denoted as a hyperparameter with I as a Identity matrix. The distribution dealt with the latent variables γ and η can be updated with weight concerning W_L (39).

$$o_J(W_L\|\gamma_L\eta) = N(W_L\|\eta\gamma_L\eta^2) \tag{34}$$

The optimal calculation of the weight concerning the various parameters defines the multiplicative diversion of η and γ_L set the diversion of the weight factor W_L derived by the forward backpropagation. It leads to achieve the inter-related dynamics for the multi-dimensional data but results in vanishing and exploding gradients during training make the network more costly in terms of parameters and computational resources. The vanishing or exploding gradient is avoided by the reduction of the parameter using Jacobean matrix [8] to preserve the gradient magnitude and protects the memory contents. It is modelled with star cell and gradient propagation unit allows to build and train Gated recurrent unit architecture of Mode RNN [38], PRED RNN [39] with the LSTM cell. It avoids the impact of long-term errors, incompressible input sequences, and bridge the temporal sequence without the sacrifice of short-time lag capabilities. The stable stacked RNN cell transformation is designed concerning gradient propagation in hidden states using equation (2). The loss (l) is computed on the basis of average of desired target prediction and can be minimized using stochastic gradient using equation (38). The stacked multiple RNN cells are derived from the equation (26) are as follows

$$(Y_H)_t^l = \sigma(W_H^{l-1} Y_I n + b_H + Z_{t-1}) \odot (W_H Y_{Ht-1}^l + b_H) \tag{35}$$

Where l-1= Hidden nodes at the lower level l= Input nodes at the current higher level The 2D lattice is derived from the temporal unfolding using depth (l) and length(T) flow in forward pass while the gradient flows in opposite direction. The gradient magnitude using eq(4) of input, weights, and the previous hidden nodes are extracted from loss moves to the output gate integrated with a Jacobean parameter.

$$\begin{aligned} \Delta W_{I,J}(T) &= \left(\frac{\partial y_k(T)}{\partial W_{I,J}}\right)^l \Delta W_{I,J}(T)\partial y_k(T)^l \\ Where &\left(\frac{\partial y_k(T)}{\partial W_{I,J}}\right)^l = J \end{aligned} \tag{36}$$

Jacobean matrix(J) using equation (36) and $\Delta W_{I,J}\partial y_k^l$ is column vector contains the partial derivatives of loss function(L) ∂L which is as nonlinear that maps the input signal X at time(t). Hidden state of the previous time step=t-1 and Y_H is the current hidden node, W is the trained weighted parameter of the cell with the input sequences of overall length (l) at T. Updated recurrence for propagation-:

$$\begin{aligned}\partial L(T)^l = \left(\frac{\partial y_k(T)}{\partial y_k(T)}\right)^{l+1} \partial L(T)^{l+1} + \\ \left(\frac{\partial y_k(T+1)}{\partial y_k(T)}\right)^{l} \partial L(T+1)^{l+1} \\ = J(T)^{l+1}\partial L(T)^{l+1} + Y_H(T+1)^l \partial L(T+1)^l\end{aligned} \tag{37}$$

Where J^l = Jacobean with respect to the input, Y_H^l = Jacobean with respect to the hidden state. Here the simple RNN cell derives the RNN to Vanilla RNN(VRNN) from the derivation of the equation (38) two Jacobeans are derived as below

$$(Y_{Ht})^l = \sigma((W_H^{l-1}Y_{In} + b_{Ih} + Z_{t-1}) \odot (W_H Y_{Ht-1}^l + bH)) \tag{38}$$

$$J^l = D\sigma(W_H^{l-1}Y_{In} + b_H + Z_{t-1}) \odot (W_H Y_{Ht-1}^l + b_H)^l W_X \tag{39}$$

$$Y_H^l = D \times \sigma(W_H^{l-1}Y_{In} + b_H + Z_{t-1}) \odot (W_H Y_{Ht-1}^l + b_H)^l W_H \tag{40}$$

Here the D denotes diagonal matrix with elements of vector x called as diagonal entities

$$\begin{aligned}J^l = D\sigma((Y_C)^l DW_X Y_O(T+1)) + D\sigma((Y_C)^l \\ D(Y_O(T+1))^l D(Y_O(T+1))^l D(Y_C(T-1))^l \\ D(Y_F((T+1))^l W_X (Y_F(T+1)) + D_N^l \\ D(YIn(T+1))^l W_X + D(YIn(T+1))^l D_N^l W_{XN})\end{aligned} \tag{41}$$

The input, forget, and activation functions are utilized to represent the gates in the previous equations.

$$\begin{aligned}Y_H^l = D\sigma((Y_C)^l DW_X Y_O(T+1)) + D\sigma((Y_C)^l \\ D(Y_O(T+1))^l D(Y_O(T+1))^l D(Y_C(T-1))^l \\ D(Y_F((T+1))^l W_X (Y_F(T+1)) + D_N^l \\ D(YIn(T+1))^l W_X + D(YIn(T+1))^l D_N^l WY_{HN}\end{aligned} \tag{42}$$

The star unit is used to begin the updated parameter. Imposed of Star unit with RNN-: The l^{th} layer of the STAR cell takes the input Y_H^l at time t from the first layer derives X_t non-linearly projects it to the space of the hidden vector h_l. In addition, the prior hidden state combines the further input into the gating parameter Y_{In}^l. The Y_F^l function, similar to the forget gate, determines how information from past hidden states and new inputs are combined to create a new hidden state, which is the STAR unit's overall dynamics[39]. The multiplicative input gate unit is introduced with Jacobeans and star parameters protect the memory contents. It relies upon the gradient computations terminated at specific points specific to the architecture, which does not impact long-term errors. A new recurrent network architecture is integrated with a robust gradient-based learning algorithm. With noisy, incompressible input sequences, it can still learn to bridge temporal encompasses longer than a thousand times without sacrificing its short-time lag capabilities.

TABLE 1. Sign language theoretical and experimental review

Sl No	Method	Contribution	Advantages	Limitations
1	Bidirectional long short-term memory fast fisher vector (FFV-BI-LSTM) [40, 41]	Training of 3D hand skeletal motion and orientation features using Leep Motion Controller	Increased Accuracy of 5% and optimization of Feature vector from 3D to 4D	Misclassification of hand motion trajectory due to minor variation or similarity with ASL words reflected in biases
2	CNN-LSTM-HMMS [42]	Multistream architecture and the joint multistream alignments proposed to create weak shape labels	Constraints in LSTM with feasible length which fits in modern GPUS, and observe significant convergence in dataset	The Dynamic weightage of the streams cannot be applied due to high computational complexity
3	Multiple Deep learning architecture [43]	3DCNN instances were used for feature learning and MLP and autoencoders used for aggregation of local features	Improve the recognition rate and accuracy	Training cost increases and not validated in the real time system
4	CNN+ BiLSTM, GAN (Generative Adversial Neural Network) [44, 45]	CNNLSTM extract pose details and GAN model to improve visual quality	The accuracy rate is 95% and framework demonstrated high human validation score in real time sign language	Challenges to handle large data, lacks in recognition accuracy and visual quality
5	Self-attention framework using vision transformer [46, 47, 48]	Tiny swin transformer model, spatial encoder and temporal module and masking using future transformer	Focused on high level semantic [65]information and eliminate redundancy	The recognition is performed only with 32 image frame sequence or video clips with 15-20 frames
6	Attention based Bidirectional LSTM with mobile Net v2 [45]	The model focus in selective crucial points, encapsulate pertinent information breakdown complication to simplify without noise	The model outperforms many contemporary sign language mechanisms validation accuracy increased up to 5%	Less comprehensive defines less than 100 words with 30 medical signs
7	MIPA-RESGCN(multi input, part attention Enhanced Residual graph convolutional network) [50]	Spatio temporal graph convolutional blocks to capture spatial and temporal relation with attention mechanisms	Significant reduction in computational complexity Efficient spatio temporal features with the removal of noise	The model fails in critical scenarios if there is a similarity of signs

Sl No	Method	Contribution	Advantages	Limitations
8	Bidirectional spatial temporal LSTM fusion attention Network [51]	CNN and spatio temporal LSTM for feature representation and uniform Neural machine translation frame work for geasture Recognition	Achieve highest accuracy while alignment results not true and feasible	Longer training time with equal aspect of testing time to have more accuracy
9	Deep CNN [52]	Deep CNN with thermal imaging capture system for the Hand gesture classification using FLIR lepton 3.5	Less space complexity and computational complexity	Challenge to recognize hand gestures in complex background variable illumination low intensity environment
10	Three multi-information sharing network (TMS-NET) [4]	End to end multistream network architecture, and multilevel sharing mechanisms	Overcome complexity and diversity of gesture motions, and makes challenges for the SLR task	Computational resources to be increased and demands training time potentially limits real world application
11	Separable parametric graph convolution(SPG-CONV) [39]	Multiple parameterized graph improve local interaction patterns handle irregular skeleton topologies	Space complexity reduced and computational cost also reduced	Fails to derive the pattern complex interactions and partitioning groups affects the recognition
12	Coherence constrained graph LSTM (CCG-LSTM) [19]	Temporal confidence gate and spatial confidence gate measuring consistency of certain motions	Group activity recognition 6.4% with improvement in accuracy and learning rate is increased with minimum loss	STCC and GCC adds complexity and reduces potential training adding parameters make model more intricate to optimize
13	Region convolutional 3D network (RC-3D) [53, 54]	The model generates candidate temporal regions jointly optimized by fusing the flow of RGB with stream network	Accurate and fast detection	Dense video captioning and localizing moments in videos is not included in the framework
14	View Adaptive neural network(VA-RNN) and View adaptive convolution neural network(VA-CNN)[55]	Consistent observation view point and skeletal transformation to the viewpoints are determined	Eliminates the influence of viewpoint enables network focuses on action features improves robustness and elevate over fitting	Requires more computational and convergence time and challenges in fusion

Sl No	Method	Contribution	Advantages	Limitations
15	Hierarchical Long short-term concurrent memory (H-LSTM) [22]	Long term interrelated dynamics stores individual motion information using new cellgate and new co-memory cells	Learns dynamic interrelated representation among multiple person in a hierarchical way to improve efficiency	Fails to infer in recognition of complex multiple persons interaction
16	Hierarchical LSTM with adaptive attention(HLSTMat) [21]	Spatial and temporal attention of specific region utilized with adaptive attention considers low level and high level visual context information	It enables more complex representation of visual data with different scales	The model is not refined to perform image captioning task
17	Skeleton joint co-attention recurrent neural networks (SC-RNN) [56]	Skeleton joint feature map is constructed with skeleton joint coattention to refine the observe motion information	Dynamically learn coattention feature map embedded with human joint and skeletal motion with new weighted gram-matrix loss	Prediction of long motions dramatically increase prediction errors and model will get collapse due to bottle neck
18	Tree structure-based traversal framework [8]	Extension in analyzation of spatial and temporal domain in the hidden sources of action related information by implementing a new gating mechanism with LSTM model	Multi-model feature fusion optimized the efficiency of methods	Complexity of traversal process increases, if maintain the adjacency information
19	3D CNN [57]	Deep convolution neural networks for transfer learning	Avoids the scarcity of large labeled dataset	Accuracy will fail if dataset is small and due to Noise [small dataset and noise limits the integrity of the model
20	3D Deep Neural attention based Bi-LSTM (hDNN-SLR) [58, 65]	Multi semantic property extraction with temporal and sequential features using 3D Deep Neural attention based Bi-LSTM	Accurate Recognition and reduction of computation overhead	Fails in continuous recognition of sign gestures and mishandling of segmentation ambiguities and moment epenthesis

Sl No	Method	Contribution	Advantages	Limitations
21	Graph convolution with attention and residual connection (GCAR) [59]	Temporal spatial features for Non skeletal points are added through Sep-TCN and joint motion is mirrored to yield the final feature vector provides discriminative short range dependencies	High Accuracy coupled with less stable computational complexity	Lack of Real time implementation and multicamera Recognition
22	Attention Based Graph Convolutional Network (GCN) [60]	Enhance Non-connected joint skeleton features with two streams Deep learning Network in which graph-based features of 47 poses using GCN, refined through attention model	Dynamic with precise efficiency and robust recognition system	Lack of evaluation of optimal number of joints results in limitation of sign words
23	CNN transfer learning [61]	Two stage CNN , one for counting predicted word and another for meaning extraction	Optimization of accuracy in the context of sign words with reduced complexity	Lack of topic collection and labels for large sentence dataset of multiple signs
24	GRU(Gated Recurrent unit) with MLP (Multi layer perceptron) SwC GR-mixer model [62]	Recognition by shifted window concatenation and temporal modeling using GRU	6.95% improvement in accuracy and overcomes the independence of single frequency dataset and avoids over fitting	Lack of interpretation in video domain cannot be implemented in Real time
25	Deep transfer learning based convolution neural network with a random forest classifier[63]	Model applied with Background Elimination and region of interest using stochastic gradient descent optimization	Reliable architecture with fine tuning provides lower learning rate with prevention of over fitting	High level parameters increases computational complexity limits to extend for low end system
26	Spatial-temporal fusion convolution Neural Network (STFC-Net)[37]	Multilevel iterative optimization using Hierarchical memory sequence network to enhance the temporal relationship	Reduction of space and time complexity of the network model	Model is over fitted due to limited SLR dataset

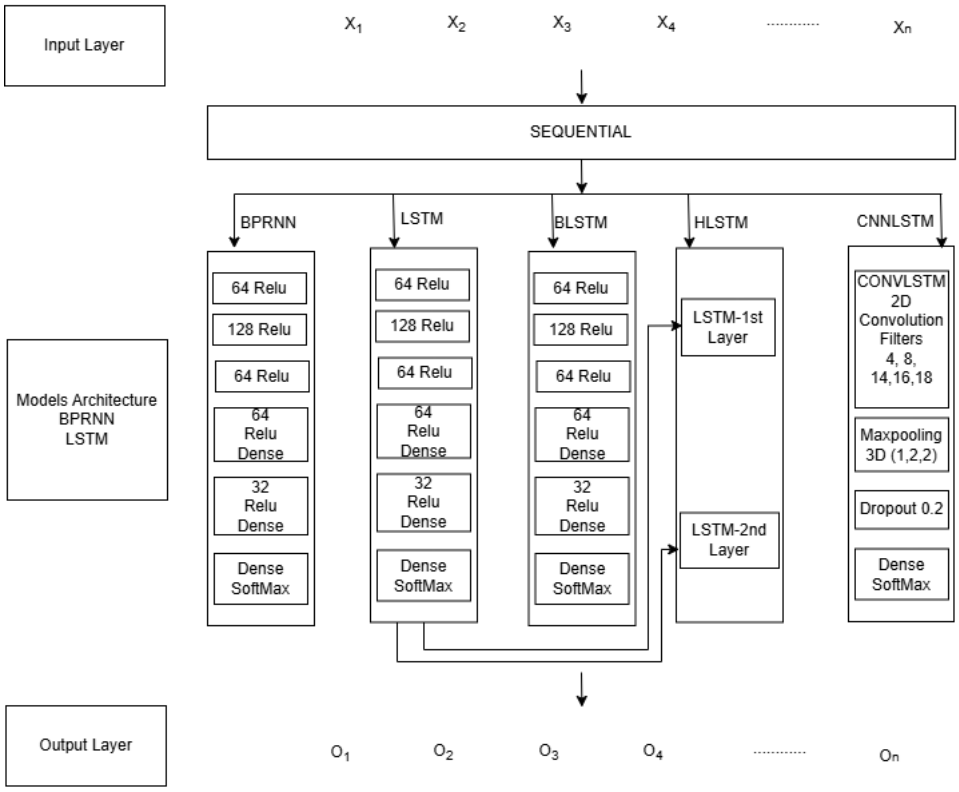
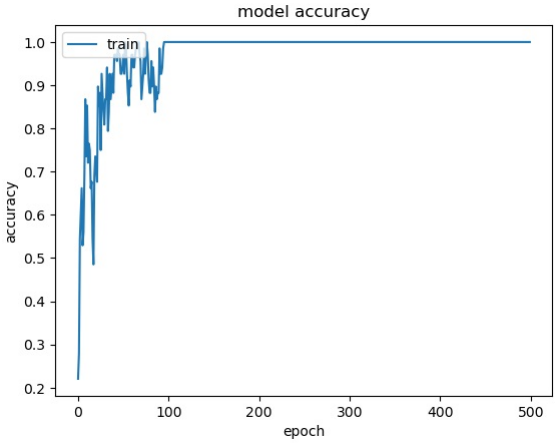


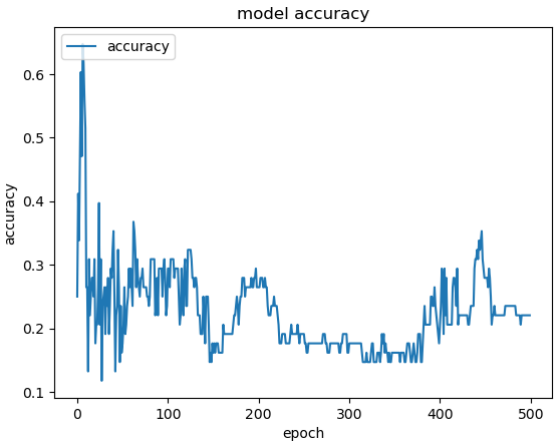
FIGURE 1. Architecture of the LSTM Models for the Sign Language Interpretation

4. Architectural and Experimental Analysis Of LSTM Models

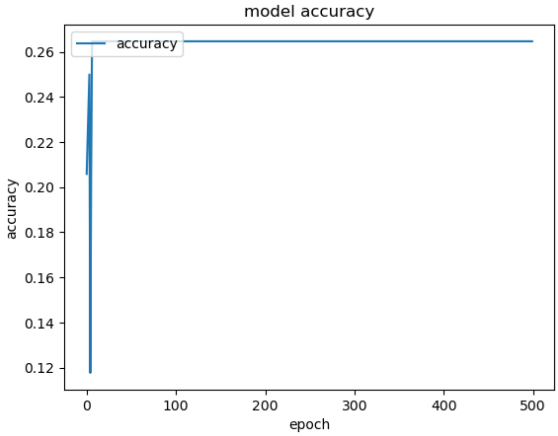
The models of LSTM [30]are evaluated with Sign language recognition system illustrated in Table 1 and experimented with the dataset of ASL by the architecture shown in Figure 1. The architecture of the deep learning models of LSTM [68] and BiLSTM, HLSTM, BaLSTM, Hierarchical Bayesian LSTM architecture using relu and softmax activation function. It is used to learn and classify sign language gestures captured from the audio and video [66] feed with the dataset. The Key features of the survey include real-time gesture detection with the analysis of accuracy and loss of recognition using new train sign language gestures. The system is built using Python, TensorFlow, OpenCV, and NumPy, making it accessible and easy to customize with the real-time sign Language Detection Using LSTM Model, Bidirectional LSTM model, Bayesian LSTM, Hierarchical Bayesian LSTM, Mode RNN, PRED RNN. The survey aims to provide a strong model analysis and define the problem due to comprehending of error from the various LSTM models. The activation functions of sequential Backpropagation Neural Network, LSTM, BiLSTM, HLSTM used hidden units, dense units with relu and soft activation function for testing and training of the sign language dataset. It is initiated using relu activation function for the first three layers with the learning model in the sequence of (64, 128, 64) and further two layers with dense model in the sequence of (64, 32). Finally, the last single layer is initiated with the dense network using the SoftMax activation function. It also deals with LSTM with convolution (CNNLSTM)[52, 64] with the architectural layer which includes the convlstm2D learning model followed by the convolution filtering mask (4, 8, 14, 16, 18) and max pooling-3D layer with the sequence of (1, 2, 2) and with the dropout of 0.2



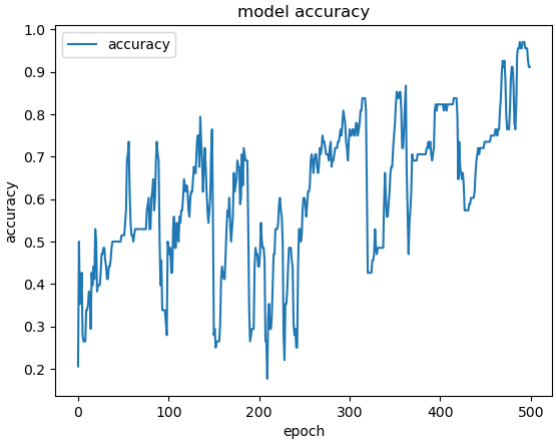
(a) RNN



(b) LSTM

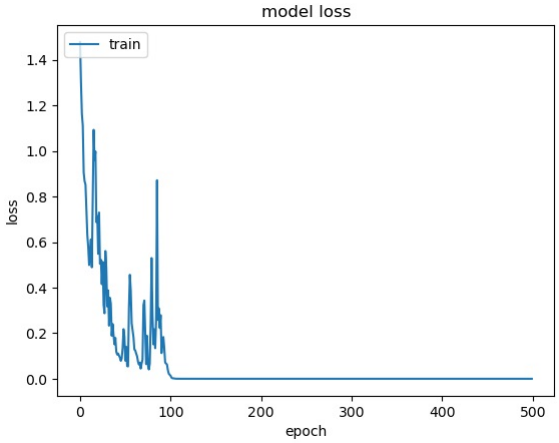


(c) BIDIRECTIONAL LSTM

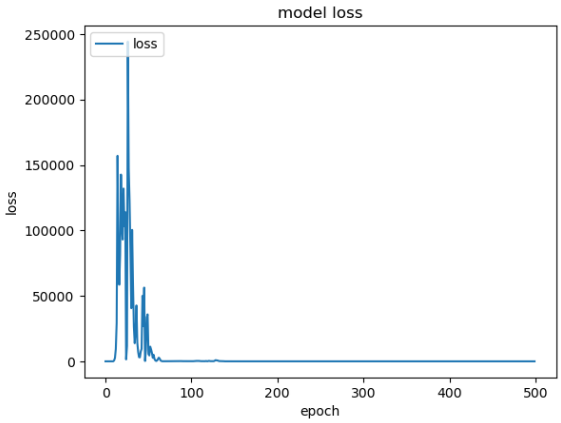


(d) HIERARCHIAL LSTM

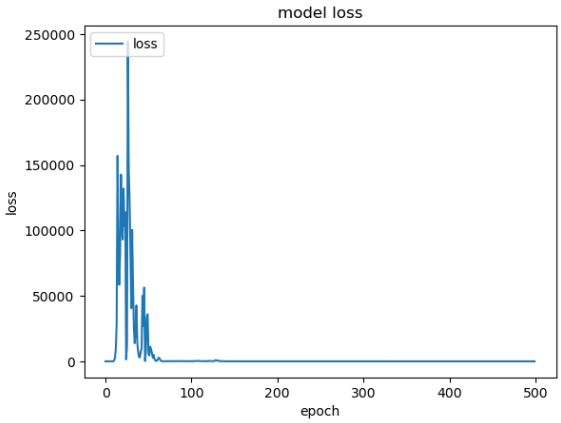
FIGURE 2. Comparative evaluation of ACCURACY of LSTM models



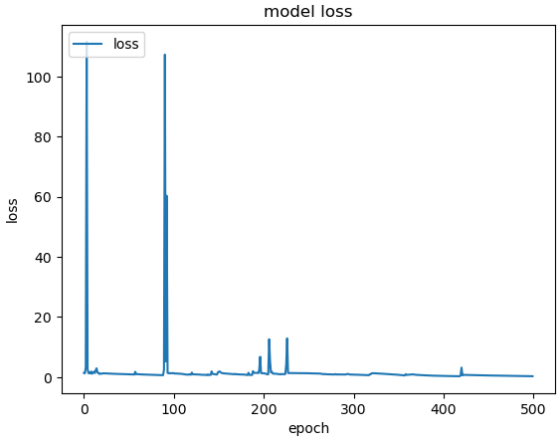
(a) RNN



(b) LSTM



(c) BIDIRECTIONAL LSTM



(d) HIERARCHIAL LSTM

FIGURE 3. Comparative evaluation LOSS of LSTM models

and at last the dense layer using SoftMax activation for the LSTM model analysis. The sign language [67] dataset is trained and tested with the above model architecture for the evaluation of the various LSTM models consisting of various units for the enhancement of gestures which makes the flow of the quality of the evaluation of the accuracy of the model in the prediction of sign images. The model emerges with the various features of the dataset Where $train_x$, $train_y$, and epochs of the model are predicted with time steps (t). The training of the LSTM models indicates whether the particular layer should behave in training mode or inference mode when the dropout or recurrent dropout is used at the time of training of the ASL dataset. The accuracy and the loss of the various LSTM models are evaluated using the following equations

$$Accuracy = TN + TP/TN + FP + TP + FN \tag{43}$$

$$LOSS = (Y - Y_{hat})^2 \tag{44}$$

In the analysis of the comparative study of various LSTM models, all the models are executed with 500 epochs to evaluate the accuracy of the models shown in Figure 2. While experimented with RNN model [40] from 0-100 epochs there is a huge variation in the accuracy of the data. After the 100 epochs, the accuracy reached the maximum of 100% whereas the LSTM model up to 10 epochs it moves from the accuracy max of 65%, and after that up to the 500 epochs the average of 30% is preserved.

In the Bidirectional LSTM model shows a constant accuracy of 26% for the entire epochs but the Hierarchical LSTM model is more dynamic and rapidly increases accuracy from lower epochs rate to higher epochs rate. The various LSTM are further evaluated to estimate the loss show in Figure 3 in which RNN model provided an increase and reduction of loss up to 100 epochs after that there is no loss in the data. Whereas LSTM model up to 50 epochs there is an average increase of loss and after that up to 80 epochs there is a consistent reduction of loss but with the Bidirectional LSTM model up to 30 epochs there is maximum to minimum loss, then after 30 epochs there is a sudden increment of loss up to 70 epochs again maximum to minimum loss is preserved after 70 epochs. While dealing with the Hierarchical LSTM model it shows that there is unique flow of loss factor compared to another model, the loss rate is increased up to 5 epochs after that there is a reduction of loss rate up to 100 epochs then there is an increase loss rate up to 105 then there is a consistent reduction in the loss rate in further epochs.

5. Conclusions

Sign language is the multidimensional interpretation of data with time series providing the effective analysis of LSTM models. The derivative of the LSTM models is analysed and interpreted with the sign language application in the context of error propagation impact on the application. The BPRNN model has been derived efficiently if the propagation of time slack factor is less than one provided optimum accuracy of 99 % and loss at average epochs in the architecture of ReLu 64, 18, 64 and SoftMax in dense layer in sign language. The multidimensional data caused an increase in exponential flow of error. The LSTM preserved CEC due to multiplicative gates provided constant error flow experimented with the model provided more accuracy of 60% at lower epoch but reduced rapidly with low loss in sign language. The conflict of weights is solved by Bidirectional LSTM optimized CEC reduced optimally the loss rate but accuracy 26% is not increased to the optimum level in sign language due to the explode of vanishing error. The multi dimensional data is effectively handled by multi level hierarchical LSTM models, avoids vanishing error and reduces the impact of long term errors, shows exponential increase in accuracy and reduction in loss for sign language. This Extensive review provided clear

interpretation of the LSTM models mathematically and experimentally with the aspects of error propagation and its impact in the multidimensional data processing in the basis of sign language interpretation. It also revealed many insightful observations such as scarcity of annotated 2D to 3D datasets while video processing. In future the review is extended to more derivative of error analysis for deep learning model with multidimensional interpretation of sign language system. To overcome the challenges of error analysis in deep learning model with the evaluation of sign language which will be helpful for the future researchers.

Acknowledgement. The authors would like to acknowledge the contribution for English recievied from Dr. Ritz Updayaya MA (English) Banaras Hindu University, Ph.D. (Indology), Leiden University, Netherlands.

References

[1] Kun, F., Junqi, J., Runpeng, C., Fei, S., Changshui, Z., (2017), Aligning where to see and what to tell: Image captioning with region-based attention and scene-specific contexts, IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(12), pp. 2321–2334.

[2] Bin, W., Zhijian, O., Zhiqiang, T., (2018), Learning transdimensional random fields with applications to language modeling, IEEE Transactions on Pattern Analysis and Machine Intelligence, 40(4), pp. 876–890.

[3] Ronald, J. W., David, Z., (1995), Gradient-based learning algorithms for recurrent networks and their computational complexity.

[4] Zhiwen, D., Yuquan, L., Junkang, C., Xiang, Y., Yang, Z., Qing, G.,(2024), Tms-net: A multi-feature multi-stream multi-level information sharing network for skeleton-based sign language recognition, Neurocomputing.., 572(3), pp. 3007-3021.

[5] Xu, Y. Z., Fei, Y., Yan, M. Z., Cheng, L. L., Yoshua, B., (2018), Drawing and recognizing chinese characters with recurrent neural network, IEEE Transactions on Pattern Analysis and Machine Intelligence, 40(4), pp. 849–862.

[6] Kyoungoh, L., Woojae, K., Sanghoon, L., (2023), From human pose similarity metric to 3d human pose estimator Temporal propagating lstm networks, IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(2), pp. 1781–1797.

[7] Minhyuk, L., Joonbum B., (2020), Deep learning based real-time recognition of dynamic finger gestures using a data glove, IEEE Access, 8, pp. 219923–219933.

[8] Jun, L., Amir, S., Dong, X., Alex C. K., Gang, W., (2018), Skeleton-based action recognition using spatio-temporal lstm network with trust gates, IEEE Transactions on Pattern Analysis and Machine Intelligence, 40(12), pp. 3007–3021.

[9] Sepp, H., Jurgen, S., (1997), Long short-term memory, Neural Computation, 9(8), pp. 1735–1780.

[10] Mostafizer, R., Yutaka, W., (2023), Multilingual program code classification using n-layered bi-lstm model with optimized hyperparameters, IEEE Transactions on Emerging Topics in Computational Intelligence, 8, pp.1452–1468.

[11] Khanh, T.P., Nguyen, K. M., Christian, G., (2022), Probabilistic deep learning methodology for uncertainty quantification of remaining useful lifetime of multicomponent systems, Reliability Engineering and System Safety, 222.

[12] Weiwen, P., Zhi, S. Y., Nan, C., (2020), Bayesian deep learning based health prognostics toward prognostics uncertainty, IEEE Transactions on Industrial Electronics, 67(3), pp. 2283–2293.

[13] Yuming, Jie, W., Yan, S., Shude, W., Zuan, F., Jie, G., (2022), Ultra-short-term interval prediction model for photovoltaic power based on bayesian optimization, Institute of Electrical and Electronics Engineers, pp. 1138–1144.

[14] Tadas, B., Chaitanya, A., Louis P. M., (2019), Multimodal machine learning: A survey and taxonomy, IEEE Transactions on Pattern Analysis and Machine Intelligence., 41(2), pp. 423–443.

[15] Vincent, L. G., Nicolas, T., (2023), Deep time series forecasting with shape and temporal criteria, IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(1), pp. 342–355.

[16] Sunghyun, S., Dohee, K., Hyerim, B., (2023), Correlation recurrent units: A novel neural architecture for improving the predictive performance of time-series data, IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(12), pp. 14266–14283.

[17] Wei, W., Yan, Y., Zhen, C., Jiashi, F., Shuicheng, Y., Nicu, S., (2019), Recurrent face aging with hierarchical autoregressive memory, IEEE Transactions on Pattern Analysis and Machine Intelligence, 41(3), pp. 654–668.
[18] Qianli, M., Sen, L., Garrison, W. C., (2022), Adversarial joint learning recurrent neural network for incomplete time series classification, IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(4), pp. 1765–1776.
[19] Jinhui, T., Xiangbo, S., Rui, Y., Liyan, Z., (2022), Coherence constrained graph lstm for group activity recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(2), pp. 636–647.
[20] Xianyun, W., Weibang, L., (2023), Time series prediction based on lstm-attention-lstm model, IEEE Access, 11, pp. 48322–48331.
[21] Lianli, G., Xiangpeng, L., Jingkuan, S., Heng-Tao, S., (2019), Hierarchical lstms with adaptive attention for visual captioning, IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 1–1.
[22] Xiangbo S., Jinhui, T., Guo, J. Q., Wei, L., Jian Y., (2021), Hierarchical long short term concurrent memory for human interaction recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence, 43(3), pp. 1110–1118.
[23] Bing, Su. and Ying Wu., (2019), Learning low-dimensional temporal representations with latent alignments, IEEE Transactions on Pattern Analysis and Machine Intelligence., pp. 1–1.
[24] Mehmet, O. T., Stefano, D. A., Jan, W., Konrad S., (2021), Gating revisited: Deep multi-layer rnns that can be trained, IEEE Transactions on Pattern Analysis and Machine Intelligence., pp. 1–1.
[25] Gers, F.A., Schmidhuber, J., Cummins, F., (1999), Learning to forget: continual prediction with lstm, In 1999 Ninth International Conference on Artificial Neural Networks, 2, pp. 850–855.
[26] Felix, A., Gers., Nicol, N. S., Jurgen, S., (2003), Learning precise timing with lstm recurrent networks, J. Mach. Learn. Res., 3, pp. 115–143.
[27] Dong, Q., William, K. C., (2023), Learning hierarchical variational autoencoders with mutual information maximization for autoregressive sequence modeling, IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(2), pp. 1949–1962.
[28] Gilmer, V., Jerome, H. F., Fei, J., Efstathios, D. G., (2022), Representational gradient boosting: Backpropagation in the space of functions, IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(12), pp. 10186–10195.
[29] David, E., Rumelhart, G., Hinton, E., Ronald, J. W., (1986), Learning representations by back-propagating errors, Nature, 323(10), pp. 533–536.
[30] Sepp, H., Jurgen, S., (1997), Long short-term memory, Neural Computation., 9(8), pp. 1735–1780.
[31] Qi, L., Jun, Z., (2015), Revisit long short-term memory: An optimization perspective.
[32] Anahita, G., Nurfadhlina, M. S., Fatimah B. S., (2024), Prediction of course grades in computer science higher education program via a combination of loss functions in lstm model, IEEE Access, 12, pp. 30220–30241.
[33] Wenzhao, Z., Jiwen, L., Jie, Z., (2023), Deep metric learning with adaptively composite dynamic constraints, IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 1–17.
[34] Minhee, K., Kaibo, L., (2021), A bayesian deep learning framework for interval estimation of remaining useful life in complex systems by incorporating general degradation characteristics, IISE Transactions, 53(3), pp. 326–340.
[35] Jun, L., Henghui, D., Amir, S., LingYu, D., Xudong, J., Gang, W., Alex, C. K., (2020), Feature boosting network for 3d pose estimation, IEEE Transactions on Pattern Analysis and Machine Intelligence., 42(2), pp. 494–501.
[36] Jie, X., Wei, Z., Fei, W., (2021), A(dp)2sgd: Asynchronous decentralized parallel stochastic gradient descent with differential privacy, IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 1–1.
[37] Cuihong X., Jingli, J., Ming, Y., Gang, Y., Yingchun G., Yuehao L., (2024), Continuous sign language recognition based on hierarchical memory sequence network, IET Computer Vision, 18(3)., pp. 247–259.
[38] Lingxiang, Y., Worapan, K., Peng, Z., Qiang, W., Jian, Z., (2023), Improving disentangled representation learning for gait recognition using group supervision, IEEE Transactions on Multimedia, 25, pp. 4187–4198.
[39] Yunbo, W., Haixu, W., Jianjin, Z., Zhifeng, G., Jianmin, W., Philip S. Y., Ming, S., (2023), Long. Predrnn: A recurrent neural network for spatiotemporal predictive learning, IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(2), pp. 2208–2225.

[40] Sunusi, B., Abdullahi, M., Kosin, C., (2022), American sign language words recognition using spatiooral prosodic and angle features: A sequential learning approach, IEEE Access, 10, pp. 15911–15923.
[41] Huangyue, Yu., Minjie, C., Yunfei, L., Feng, L., (2023) First and third person video coanalysis by learning spatial-temporal joint attention, IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(6), pp. 6631–6646.
[42] Oscar, K., Necati, C. C., Hermann, N., Richard, B., (2020), Weakly supervised learning with multi-stream cnn-lstm-hmms to discover sequential parallelism in sign language videos, IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(9), pp. 2306– 2320.
[43] Muneer, A., Ghulam, M., Wadood, A., Mansour, A., Mohammed, A. B., Tareq, S. A., Hassan, M., Mohamed, A. M., (2020), Deep learning-based approach for sign language gesture recognition with efficient hand gesture representation, IEEE Access, 8, pp. 192527–192542.
[44] Natarajan, B., Rajalakshmi, E., Elakkiya, R., Ketan, K., Ajith A., Lubna, A. G., Subramaniyaswamy, V., (2022), Development of an end-to-end deep learning framework for sign language recognition, translation, and video generation, IEEE Access, 10, pp. 104358–104374.
[45] Amimul, I., Abrar, F. E., Lutfun, N., Muhammad, A. Kadir., (2024), Medisign: An attention-based cnn-bilstm approach of classifying word level signs for patient doctor interaction in hearing impaired community, IEEE Access, 12, pp. 33803–33815.
[46] Yao, D., Pan, X., Mingye, W., Xiaohui, H., Zheng, Z., Jiaqi, L., (2022), Full transformer network with masking future for word-level sign language recognition, Neurocomputing, 500(8), pp. 115–123.
[47] Gaspard, H., Jong, W. K., Beakcheol, J., (2022), A multi-headed transformer approach for predicting the patient's clinical time-series variables from charted vital signs, IEEE Access, 10, pp. 105993–106004.
[48] Lipisha, C., Tejaswini, A., Enjamamul, H., Ifeoma, N., (2023), Signnet ii: A transformer-based two-way sign language translation model, IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(11), pp. 12896–12907.
[49] Yan, H., Qi, W., Wei, W., Liang, W., (2018), Image and sentence matching via semantic concepts and order learning, IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(3), pp. 636–650.
[50] Neelma, N., Hasan, S., Sara, A., Osman, H., Muhammad, K. E., (2023), Miparesgcn: a multi-input part attention enhanced residual graph convolutional framework for sign language recognition, Computers and Electrical Engineering, 112(12).
[51] Qinkun, X., Xin, C., Xue, Z., Xing L., (2020), Multi-information spatial temporal lstm fusion continuous sign language neural machine translation, IEEE Access, 8, pp. 216718– 216728.
[52] Daniel, S. B., Aveen, D., Ajit, J., Phaneendra, K. Y., OmJee, P., Linga, R. C., (2021), Robust hand gestures recognition using a deep cnn and thermal images, IEEE Sensors Journal, 21(12), pp. 26602–26614.
[53] Huijuan, X., Abir, D., Kate, S., (2019), Two-stream region convolutional 3d network for temporal activity detection, IEEE Transactions on Pattern Analysis and Machine Intelligence, 41(10), pp. 2319–2332.
[54] Hamzah, L., (2022), An efficient two-stream network for isolated sign language recognition using accumulative video motion, IEEE Access, 10, pp. 93785–93798.
[55] Pengfei, Z., Cuiling, L., Junliang, X., Wenjun, Z., Jianru, X., Nanning, Z., (2019), View adaptive neural networks for high performance skeleton-based human action recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence, 41(8), pp. 1963–1978.
[56] Haocong, R., Siqi, W., Xiping, H., Mingkui, T., Yi, G., Jun, C., Xinwang, L., Bin, H., (2022), A self-supervised gait encoding approach with locality awareness for 3d skeleton based person re-identification, IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(10), pp. 6649–6666.
[57] Muneer, A., Ghulam, M., Wadood, A., Mansour, A., Mohamed, A. B., Mohamed, A. M., (2020), Hand gesture recognition for sign language using 3dcnn, IEEE Access., 8, pp. 79491–79509.
[58] Rajalakshmi, E., Elakkiya, R., Subramaniyaswamy, V., Prikhodko Alexey, L., Grif, M., Maxim, B., Ketan, K., Lubna, A. G., Ajith, A., (2023), Multi- semantic discriminative feature learning for sign gesture recognition using hybrid deep neural architecture, IEEE Access, pp. 2226–2238.
[59] Abu, S. M., Mehedi, H. A., Satoshi, N., Jungpil S., (2024), Sign language recognition using graph and general deep neural network based on large scale dataset. IEEE Access, 12, pp. 34553–34569.
[60] Jungpil, S., Abu, S., Musa, M., Kota, S., Koki, H., Mehedi, H. A., (2023), Dynamic korean sign language recognition using pose estimation based and attention-based neural network, IEEE Access, 11, pp. 143501–143513.

[61] Tamer, S., (2023), Two-stage deep learning solution for continuous arabic sign language recognition using word count prediction and motion images, IEEE Access., 11, pp. 126823– 126833.
[62] Tianyu, L., Tangfei, T., Yizhe, Z., Min, L., Jieli, Z., (2024), A signer independent sign language recognition method for the single-frequency dataset. Neurocomputing, 582(5).
[63] Sunanda, D., Samir, I., Nieb, H. N., Nazmul, S., Hui, W., (2023), A hybrid approach for bangla sign language recognition using deep transfer learning model with random forest classifier, Expert Systems with Applications, 213(3).
[64] Tafia, H. P., et. al., (2024), Fine-Tuning of Predictive Models CNN-LSTM and CONV-LSTM for Nowcasting PM2.5Level, IEEE Access, 12, pp. 28988-29003.
[65] Yan, H., Qi, W., Wei, W., Liang, W., (2020), Image and Sentence Matching via Semantic Concepts and Order Learning, IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(3), pp. 636-650.
[66] Mucahit, E. Y. M., Suleyman, E., (2022), BabyPose: Real-Time Decoding of Baby's Non-Verbal Communication Using 2D Video-Based Pose Estimation, IEEE Sensors Journal, 22(14), pp. 13776-13784
[67] Anis, K., et. al., (2020), A Novel Geometric Framework on Gram Matrix Trajectories for Human Behavior Understanding, IEEE Transactions on Pattern Analysis and Machine Intelligence., 42 (1), pp. 1-14.
[68] Tresa, J., Bindiya, T. S., (2023), Realization and hardware implementation of gating units for long short-term memory network using hyperbolic sine functions, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 42(12), pp. 5141–5145.

Harapriya Kar Graduate Student Member IEEE, holds a B.Tech from Bijupattanaik University of Technology (2020) and an M.Tech from Pondicherry University (2022) in Computer Science and Engineering. Currently pursuing her Ph.D. at Vellore Institute of Technology, she specializes in deep learning and image processing for sign language. Her expertise spans data science, deep learning, image processing, NLP, ML, and AR/VR. Notable achievements include an AR/VR award from IIT Bhubaneswar (2019) and academic excellence as a batch topper in 2018 and 2020. Her research focuses on applying deep learning techniques to sign language interpretation.

Dr. Viswanathan P Senior Member, IEEE/ACM, is a Professor at the School of Computer Science and Engineering, Vellore Institute of Technology, India. He earned his D.E. from VIT in 2014. His research focuses on Digital Image Processing, Machine Learning, Cloud Computing, IoT, and Deep Learning. He has received multiple accolades including the Indian Science Congress Best Poster Award (2007) and VIT Most Active Researcher Award (2010-2022). He holds patents in Agricultural and Forensic applications of Machine Learning, demonstrating significant contributions to these fields.