event observed at time instant t and hence is sensitive to the observed input $I'(t)$ than the event layer, which is more persistent across events. The event layer is a gated layer, which receives input from the encoder as well as the recurrent event model. However, the inputs to the event layer are modulated by a self-supervised gating signal (Section 3.3), which is indicative of the quality of predictions made by the recurrent model. The gating allows for updating the weights quickly but also maintains a coherent state within the event.

Why recurrent networks? While convolutional decoder networks [17] and mixture-of-network models [37] are viable alternatives for future prediction, we propose the use of recurrent networks for the following reasons. Imagine a sequence of frames $I_a = (I_a^1, I_a^2, \dots I_a^n)$ corresponding to the activity a. Given the complex nature of videos such as those in instructional or sports domains, the next set of frames can be followed by frames of activity b or c with equal probability, given by $I_b = (I_b^1, I_b^2, \dots I_b^m)$ and $I_c = (I_c^1, I_c^2, \dots I_c^k)$ respectively. Using a fully connected or convolutional prediction unit is likely to result in the prediction of features that tend to be the average of the two activities a and b, i.e. $I_{avg}^k = \frac{1}{2}(I_b^k + I_c^k)$ for the time k. This is not a desirable outcome because the predicted features can either be an unlikely outcome or, more probably, be outside the plausible manifold of representations. The use of recurrent networks such as RNNs and LSTMs allow for multiple futures that can be possible at time $t+1$, conditioned upon the observation of frames until time t.

3.1.3 Feature Reconstruction

In the proposed framework, the goal of the perceptual processing unit (or rather the reconstruction network) is to reconstruct the predicted feature y'_{t+1} given a source prediction h_t, which maximizes the probability

$$p(y'_{t+1}|h_t) \propto p(h_t|y'_{t+1})\, p(y'_{t+1}) \tag{2}$$

where the first term is the likelihood model (or translation model from NLP) and the second is the feature prior model. However, we model $\log\, p(y'_{t+1}|h_t)$ as a log-linear model $f(\cdot)$ conditioned upon the weights of the recurrent model ω_p and the observed feature $I'(t)$ and characterized by

$$\log\, p(y'_{t+1}|h_t) = \sum_{n=1}^{t} f(\omega_p, I'(t)) + log\, Z(h_t) \tag{3}$$

where $Z(h_t)$ is a normalization constant that does not depend on the weights ω_p. The reconstruction model completes the generative process for forecasting the feature at time $t+1$ and helps in constructing the self-supervised learning setting for identifying event boundaries.

3.2. Self-Supervised Learning

The quality of the predictions is determined by comparing the prediction from the predictor model $y'(t)$ to the observed visual feature $I'(t)$. The deviation of the predicted input from the observed features is termed as the perceptual prediction error $E_P(t)$ and is described by the equation:

$$E_P(t) = \sum_{i=1}^{n} \|I'(t) - y'(t)\|_{\ell_1}^2 \tag{4}$$

where $E_P(t)$ is the perceptual prediction error at time t, given the predicted visual $y'(t)$ and the actual observed feature at time t, $I'(t)$. The predicted input is obtained through the inference function defined in Equation 2. The perceptual prediction error is indicative of the prediction quality and is directly correlated with the quality of the recurrent model's internal state $h(t)$. Increasingly large deviations indicate that the current state is not a reliable representation of the observed event. Hence, the gating signal serves as an indicator of event boundaries. The minimization of the perceptual prediction error serves as the objective function for the network during training.

3.3. Error Gating for Event Segmentation

The gating signal (Section 3) is an integral component in the proposed framework. We hypothesize that the visual features of successive events differ significantly at event boundaries. The difference in visual features can be minor among sub-activities and can be large across radically different events. For example, in Figure 1, we can see that the visual representation of the features learned by the encoder network for the activities *take bowl* and *crack eggs* are closer together than the features between the activities *take bowl* and *spoon flour*. This diverging feature space causes a transient increase in the perceptual prediction error, especially at event boundaries. The prediction error decreases as the predictor model adapts to the new event. This is illustrated in Figure 1. We show the perceptual prediction error (second from the bottom) and the ground truth segmentation (second from the top) for the video Make Pancake. As illustrated, the error rates are higher at the event boundaries and lower among "in-event" frames.

The unsupervised gating signal is achieved using an anomaly detection module. In our implementation, we use a low pass filter. The low pass filter maintains a relative measure of the perceptual prediction error made by the predictor module. It is a relative measure because the low pass filter only maintains a running average of the prediction errors made over the last n time steps. The perceptual quality metric, P_q, is given by:

$$P_q(t) = P_q(t-1) + \frac{1}{n}(E_P(t) - P_q(t-1)) \tag{5}$$