Intentional Evolutionary Learning for Untrimmed Videos with Long Tail Distribution

Yuxi Zhou[1,2,*†], Xiujie Wang[1,*], Jianhua Zhang[1,†], Jiajia Wang[1], Jie Yu[1], Hao Zhou[1], Yi Gao[1], Shengyong Chen[1]

[1]Department of Computer Science, Tianjin University of Technology, Tianjin, China
[2]DCST, BNRist, RIIT, Institute of Internet Industry, Tsinghua University, Beijing, China
joy_yuxi@pku.edu.cn, 1747897282@qq.com, zjh@ieee.org, wjj21@stud.tjut.edu.cn, YJ15303364087@gmail.com, zhouhao@stud.tjut.edu.cn, gaoyi01020304@stud.tjut.edu.cn, sy@ieee.org

Abstract

Human intention understanding in untrimmed videos aims to watch a natural video and predict what the person's intention is. Currently, exploration of predicting human intentions in untrimmed videos is far from enough. On the one hand, untrimmed videos with mixed actions and backgrounds have a significant long-tail distribution with concept drift characteristics. On the other hand, most methods can only perceive instantaneous intentions, but cannot determine the evolution of intentions. To solve the above challenges, we propose a loss based on Instance Confidence and Class Accuracy (ICCA), which aims to alleviate the prediction bias caused by the long-tail distribution with concept drift characteristics in video streams. In addition, we propose an intention-oriented evolutionary learning method to determine the intention evolution pattern (from what action to what action) and the time of evolution (when the action evolves). We conducted extensive experiments on two untrimmed video datasets (THUMOS14 and ActivityNET v1.3), and our method has achieved excellent results compared to SOTA methods. The code and supplementary materials are available at https://github.com/Jennifer123www/UntrimmedVideo.

Introduction

Humans are born with the ability to observe the world and understand the intentions of the collaborators (i.e., predict what will happen soon). The ability to understand intention is fundamental to interaction between human and environment. However, designing algorithms to automatically understand intentions (Wanyan et al. 2023) is challenging, as it is necessary to model the relationship between past and future events without completely observing untrimmed videos.

Currently, most methods on intent understanding primarily focus on trimmed videos, which process a full video into short clips with one action label each and make it unsuitable for direct application in real-world scenarios. This is due to the fact that videos are generally untrimmed. Recently, (Rodin et al. 2022) attempted to fine-tune trimmed methods to untrimmed videos and concluded that "performing action prediction tasks on untrimmed videos is challenging". We believe that one reason for the unsatisfactory results is that it ignores the long-tail distribution. Becasue in untrimmed vidoes, human actions coexisting with noisy backgrounds and thus each category of actions constitutes a minority among the background samples which lead to a long-tail distribution phenomenon. Taking the "Cliff Diving" video in the THUMOS14 dataset in Figure 1 as an example, the video lasts for 6 minutes, with a few target actions "diving" and "cliff diving" alternating with messy backgrounds. Moreover, as untrimmed videos manifest as video data streams in natural scenarios, the distribution differences between current data streams and new data streams may be substantial. However, there is no relevant method to solve the adaptive problem of long-tail distribution with concept drift (Krawczyk et al. 2017) characteristics (which refers to the possibility that data from non-stationary models may evolve over time, resulting in changes in target concepts and/or attribute distributions) in video streams.

Additionally, existing intent understanding efforts can only predict the subsequent action but fail to assess the persistence and evolution patterns of intentions. However, predicting the evolution patterns of intentions can provide very important support for human-machine collaboration. For example, in Figure 1, predicting the current action A4 is 'cliff-diving' and how long the person is about to 'dive' is important for warning of dangerous scenarios.

To address these challenges, a novel intentional evolutionary learning method is developed. Our work and contributions can be summarized as follows.

- A loss based on Instance Confidence and Class Accuracy (ICCA) is presented that significantly enhances the classification accuracy under the influence of long-tail distribution with concept drift characteristics.
- An intention-oriented evolutionary learning method is proposed to determine the intention evolution pattern (from what action to what action) and the time of evolution (when the action evolves).
- We demonstrate the effectiveness and advancement of our proposed method on THUMOS14 and ActivityNET v1.3 datasets.

*These authors contributed equally.
†Corresponding authors.

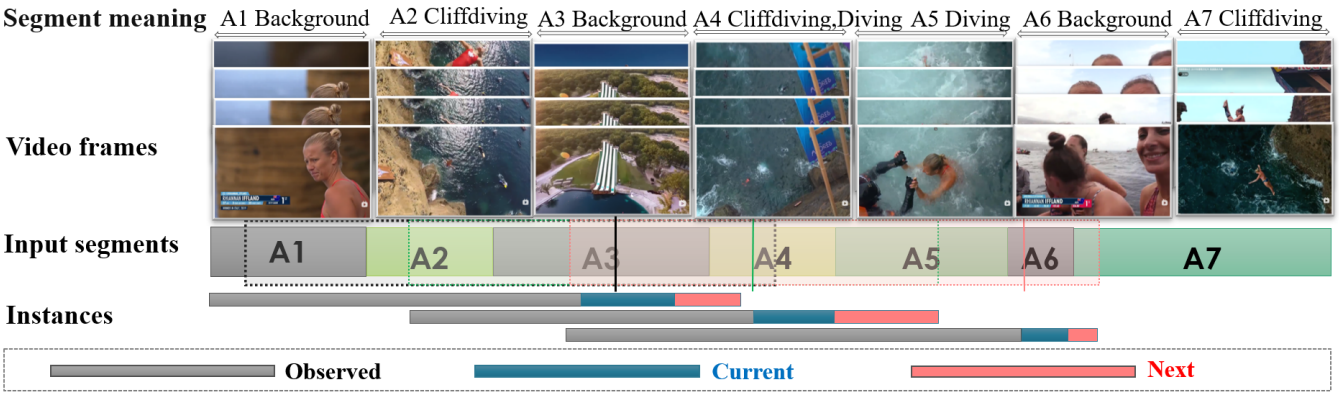


Figure 1: An example of untrimmed video which are long-tail distribution, as actions are alternating with many messy backgrounds. If human intent can be predicted as early as possible, it can provide auxiliary support for human-machine collaboration.

Related Work

Intention Understanding based on Trimmed Videos

Trimmed-video-based intent understanding covers a wide range of tasks, including traffic prediction (Bai et al. 2020), anomaly detection (Liu et al. 2021), human behavior intent prediction (Wanyan et al. 2023), etc. Among them, human behavior intent prediction tasks are getting closer to the real intents of people with the advancement of algorithms. Recent years, (Zheng et al. 2023) predicted what the next action would be as early as possible.

Above tasks are all based on trimmed video datasets and only make instantaneous judgments, but cannot determine evolution of intention. Duration estimation studies have been done in traffic(Grigorev et al. 2022) and medical(Bodenstedt et al. 2019) fields, but often need other modalities' support. Based on single video modality, (Abu Farha, Richard, and Gall 2018) achieved long-term prediction of activity sequences, as well as the start and end times of each activity. However, it was based on trimmed videos and directly used ground truth labels, which is not in line with real-world application scenarios. We study the estimation of motion duration based on untrimmed videos to guide sustained intention understanding.

Intention Understanding based on Untrimmed Videos

Currently, deep learning is less applied on untrimmed videos. (Gao, Yang, and Nevatia 2017) proposed an enhanced RED network for action prediction, which uses reinforcement learning to encourage early and correct predictions. (Ke, Fritz, and Schiele 2019) proposed an attentive temporal feature, which used multi-scale temporal convolution to process temporal-conditioned observations. (Wang et al. 2021) proposed TTPP framework, reusing the Transformer-style architecture to aggregate observed features and then using a lightweight network to progressively predict future features and actions.

Recently, (Rodin et al. 2022) tried to fine-tune trimmed methods for untrimmed videos for action prediction, but got poor results. We argue they ignored the long-tail distribution what is common in untrimmed videos. Specifically, untrimmed videos usually appear as streams, and the distribution difference between current data streams and new data streams may be very large. At present, there is no relevant method to solve the self-adaptive problem of long-tail distribution with drift characteristics in video datasets.

Long-tail Distribution

Long-tail distribution is a basic problem, especially for real-world deployment. Usually, weighting strategy is employed to deal with these problems by using weights to measure the penalty caused by prediction errors on samples.

For example, focal Loss (Lin et al. 2017) focuses more on fewer and harder samples, assigning them higher weights. EQLv1 (Tan et al. 2020) tried to provide a better weight allocation system, using class frequency to assign sample weights. EQLv2 (Tan et al. 2021) further improved, providing smoother constraints using the gradient of class frequency. CDB loss (Sinha, Ohashi, and Nakamura 2022) dynamically measures the instantaneous difficulty of each class during the model training. Considering that untrimmed videos are presented as video data streams in natural scenes, the distribution difference between current and new data streams may be very large. We propose ICCA loss to specifically address the adaptive problem of long-tail distribution with concept drift characteristics in video streams.

Methodology

Problem Definition

Given a series of untrimmed streaming videos $V = \{v_1, v_2, v_3, ..., v_M\}$, we use a sliding window strategy to expand the data, and obtain output $S_c = \left\{\hat{y}, \hat{y}', \tilde{D}\right\}$ through the intention prediction model, where $\hat{y} \in R^{\mathbb{C}+1}$ represents the potential action, which is the last category label recognized in the observation video frames, $\hat{y}' \in R^{\mathbb{C}+1}$ represents the evolution action, which is the predicted next category label, $\tilde{D}$ represents the remaining duration of the predicted

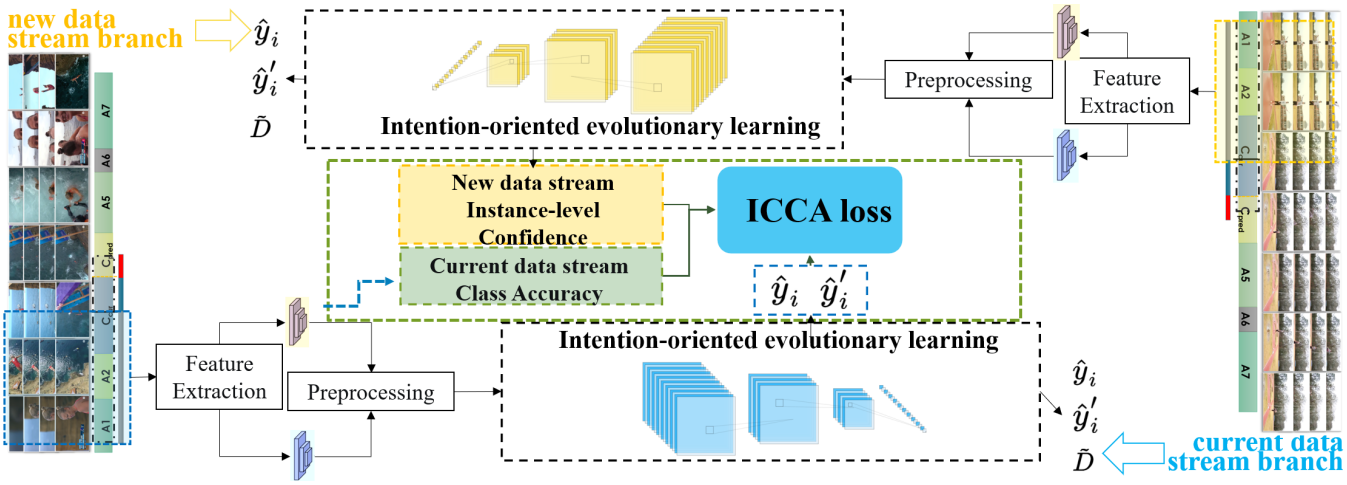


Figure 2: The overall framework.

current action. Notice that $\mathbb{C}$ is the number of action categories and $\mathbb{C}+1$ represents all categories including background. The current data stream is used as our training data, and the new data stream is used as our testing data.

Sliding window. Sliding window is a common data augmentation strategy. For detailed settings, please refer to the supplementary materials.

Framework Overview

The overall framework diagram of our method can be seen in Figure 2. It can be roughly divided into two main modules: the ICCA loss module and the intention-oriented evolutionary learning module.

For the blue current data stream, the "observation part" of the sliding window is taken as input. After feature extraction and preprocessing, video spatio-temporal features and duration information are obtained. These are put into our intention-oriented evolutionary learning model and constrained by ICCA loss to obtain the intention prediction triplet result $S_c = \left\{\hat{y}, \hat{y}', \tilde{D}\right\}$ (the process of the new data stream branch is basically the same). Specifically, our model can dynamically sense the distribution of the new data stream. The most recent new data stream result provides instance-level confidence and the previous epoch of the current round provides class accuracy. These two indicators are passed into the current data stream branch to guide ICCA loss constraints.

Loss based on Instance Confidence and Class Accuracy (ICCA loss)

The current SOTA method for solving the long-tail distribution problem is CDB-W loss(Sinha, Ohashi, and Nakamura 2022), which dynamically measures the instantaneous difficulty of each class during the model training phase. However they introduced a class-balanced subset, which cannot dynamically perceive the data distribution of new data streams. We propose *Instance Confidence and Class Accuracy (ICCA loss)* to solve the self-adaptation problem of the long-tail distribution with the concept drift characteristics in the untrimmed video.

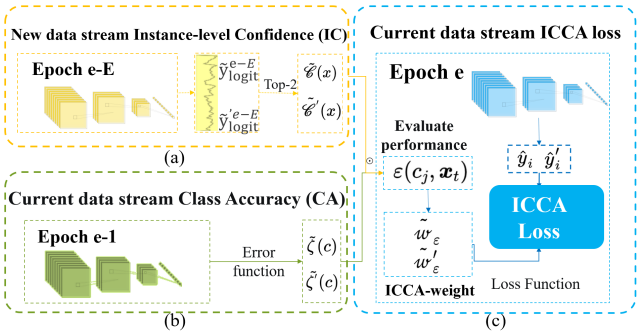


Figure 3: The specific implementation of ICCA loss in our method. (a) The output of the recent new data stream in the epoch $e-E$, obtaining Instance-level Confidence. (b) Output of the current data streams in the previous epoch $e-1$, obtaining Class Accuracy. (c) The result of combining (a) and (b) in the current epoch e to obtain evaluation indicators and ICCA weight, and finally obtain ICCA loss constraints.

Instance-level confidence. For a video sample, when the predicted confidence of a certain category is higher, the difference in predicted probability between the category with the highest predicted probability and the second highest is generally larger. Therefore, we use the difference in probability between the model's first and second highest predicted categories to approximate the measure of the model's confidence. Formally, the predicted confidence of the model on sample $x_t^{(i)}$ in the t-th batch is defined as:

$$\mathscr{C}\left(x_t^{(i)}\right) = P\left(\dot{\hat{y}}_t^{(i)} \mid x_t^{(i)}\right) - P\left(\ddot{\hat{y}}_t^{(i)} \mid x_t^{(i)}\right), \quad (1)$$

where $\hat{y}_t^{(i)}$ represents the predicted value of the model on the sample $x_t^{(i)}$, $\dot{\hat{y}}_t^{(i)}$ and $\ddot{\hat{y}}_t^{(i)}$ represent the classes with the highest and second highest predicted probabilities of the model on the sample $x_t^{(i)}$, respectively.

The approximate measure of confidence we defined above does not require prior knowledge of the ground truth of the videos. It only requires the use of a model trained to a certain extent to obtain the predicted logits of the classes and thus obtain the confidence.

Class accuracy. We have empirically found that the predicted performance of the classes in the $(e-1)$-th epoch of the data stream has an important guiding role for the weight distribution of the classes in the e-th epoch. We use $\zeta(c_j)$ to represent the probability of correctly predicting the c_j-th class, where $\zeta(c_j)$ can be any error function.

It has been found through experiments that using the accuracy of each class is better, so the $\zeta(c_j)$ obtained using the accuracy of each class can be expressed as:

$$\zeta\left(c_j\right)=\frac{\mathrm{TP}\left(c_j\right)}{\mathrm{TP}\left(c_j\right)+FP\left(c_j\right)}, \tag{2}$$

where $TP(c_j)$ is true positive and $FP(c_j)$ is false positive.

ICCA loss. In the e-th epoch, we use the instance-level confidence $\mathscr{C}\left(x_t^{(i)}\right)$ obtained in the most recent new data stream and the average class accuracy $\zeta(c_j)$ obtained in the previous epoch to dynamically evaluate the performance measure of each class. We use the product of the predicted confidence of the model on sample $x_t^{(i)}$ and the probability of correctly predicting class c_j by the model $\mathscr{C}\left(\boldsymbol{x}_t^{(i)}\right)\cdot\zeta\left(c_j\right)$ to approximate the probability that the model predicts correctly when the predicted class is c_j on the new data stream sample $x_t^{(i)}$. Conversely, its predicted error probability is $\mathscr{C}\left(\boldsymbol{x}_t^{(i)}\right)\cdot\left(1-\zeta\left(c_j\right)\right)$.

Formally, the performance measure of the model for class c_j on the data stream set x_t can be defined as

$$\begin{aligned}\varepsilon\left(c_j,\boldsymbol{x}_t\right)=&\sum_{i=1}^{N}\frac{\mathscr{C}\left(\boldsymbol{x}_t^{(i)}\right)\cdot\zeta\left(c_j\right)}{\left|\left\{i\mid\dot{y}_t^{(i)}=c_j\right\}\right|}\\&+\sum_{i=1}^{N}\frac{\mathscr{C}\left(\boldsymbol{x}_t^{(i)}\right)\cdot\left(1-\zeta\left(c_j\right)\right)}{\left|\left\{i\mid\dot{y}_t^{(i)}\neq c_j\right\}\right|}.\end{aligned} \tag{3}$$

Referring to the constraint of CDB-W loss, we use ${}_e\left(c_j,\boldsymbol{x}_t\right)$ to represent the weight corresponding to class c_j

$$w_\varepsilon\left(c_j,\boldsymbol{x}_t\right)=\left|\underset{i\in\mathbb{C}+1}{\Omega}(\varepsilon\left(c_i,\boldsymbol{x}_t\right))-\varepsilon\left(c_j,\boldsymbol{x}_t\right)\right|, \tag{4}$$

where $C+1$ represents the total number of all classes including the background class, and Ω represents operations such as taking the maximum value, taking the average value, summing, etc. $|*|$ represents taking the absolute value.

To explain the weighted loss based on ICCA, we use the most traditional cross-entropy loss here. Thus, the weighted cross-entropy loss based on ICCA is calculated as:

$$\begin{aligned}ICCA\_loss_{ce}&=-\sum_{i=0}^{C}w_\varepsilon\left(c_j,\boldsymbol{x}_t\right)y_i\log\left(p_i\right)\\&=-w_\varepsilon\left(\boldsymbol{c}_j,\boldsymbol{x}_t\right)log\left(p_c\right).\end{aligned} \tag{5}$$

The specific implementation of ICCA loss. We propose ICCA loss to mitigate the inherent problem of long-tail distribution (that is, for untrimmed videos, the action class is a minority class compared to the interspersed background class, but the background class itself does not have common characteristics) with concept drift. Specifically, our ICCA loss can make the distribution of the current data stream as close as possible to the distribution of the new data stream without knowing the distribution of the new data stream in advance, thereby effectively alleviating the problem of long-tail distribution with concept drift properties caused by inconsistent distribution between current data stream and new data stream.

The ICCA loss constraint of the model is shown in Figure 3. We use e to represent the current epoch, $e-1$ to represent the previous epoch, and E to represent the fusion frequency of the new data streams, that is, every E epochs, the the new data streams is used to obtain results based on the new data streams (the ablation experiment will explore the fusion frequency later).

Figure 3(a) shows the results of applying the current model to the new data stream at the $(e-E)$-th epoch. Here, we use $\tilde{y}_{logit}^{e-E}$ and $\tilde{y'}_{logit}^{e-E}$ to represent the recognition probability scores and prediction probability scores of each class obtained by the new data stream, respectively. Then, based on the two probability scores, we obtain the class confidence of recognition and prediction based on the new data stream $\tilde{\mathscr{C}}(x)$ and $\tilde{\mathscr{C}}'(x)$ (using Equation (1)).

Figure 3(b) shows the current data stream results at the $(e-1)$-th epoch. We use the accuracy of each class as the evaluation indicator and obtain the accuracy of each class for recognition and prediction of the current data stream at the $\tilde{\zeta}(c)$ and $\tilde{\zeta}'(c)$ according to Equation (2).

In Figure 3(c), the two guided performance indicators in Figure 3(a) and (b) are combined with the recognition and prediction results of the current model to calculate the ICCA loss of the current epoch using Equations(3) (4) and (5).

Intention-oriented Evolutionary Learning

We define intention interpretation as action evolution learning guided by potential intentions and potential actions, constrained by intention coherence in the intention semantic space, so that the predicted intention gradually approaches the true intention. Unlike previous work (Girdhar and Grauman 2021; Wanyan et al. 2023), which only judges an instantaneous concept (can only predict what the future intention is), our method can judge the evolution pattern (from what action to what action) and evolution timing (when to evolve) of the intention.

Data preprocessing. Following (He et al. 2022), we use the I3D (Carreira and Zisserman 2017) model to extract video features based on a sliding window to obtain the feature matrix $F\in R^{\mathbb{B}\times\mathbb{D}}$, where $\mathbb{B}$ is the batch size and $\mathbb{D}$ is the feature dimension.

Considering that the feature matrix F is the most primitive feature representation of the video stream V, we use a fully connected function Ψ_F to obtain the initialized latent action $\hat{y}$ representing the most likely class of the current

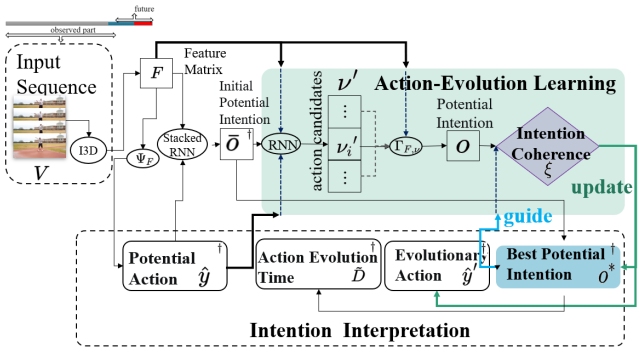


Figure 4: The architecture of the intention-oriented evolutionary learning. The † in the upper right corner indicates that the variable is an output.

video segment.

Through the potential actions $\hat{y}$ and feature matrix F, we use $StackedRNN$ to obtain the initialized potential intention $\bar{o} \in R^{\mathbb{B}\times(\mathbb{D}+1)}$. The initialized potential intention $\bar{o}$ combines both the original video features and the current most likely category semantics to generate a preliminary intention interpretation. In addition, we initialize the initialized potential intention $\bar{o}$ as the best potential intention o^*.

By combining the feature matrix F, potential action $\hat{y}$ and initialized potential intention $\bar{o}$, the possible action candidate list $\nu' = [..., \nu'_i, ...]$ is obtained through RNN ,

$$\nu' = RNN(F, \hat{y}, \bar{o}) \in R^{\mathbb{B}\times(\mathbb{D}+1)\times\mathbb{N}}, \tag{6}$$

where $\mathbb{N}$ is the number of candidate actions in the action candidate list ν'. Next, we will traverse the action candidate list ν' to implement action evolution learning and iteratively evolve and update the best potential intention o^* and evolutionary actions $\hat{y}'$.

Action evolution learning. Take a candidate action ν'_i from the action candidate list $\nu' = [..., \nu'_i, ...]$ $(i < \mathbb{N})$, combine it with the feature matrix F, execute the $\Gamma_{F,\nu}$ function to obtain the latent intent $o \in R^{\mathbb{B}\times(\mathbb{D}+1)}$. The meaning of the potential intention o here is the intention based on the semantic relationship of the candidate action ν'_i) and the original feature F. This intention may be closer to the real intention or further away from the real intention. The best potential intention o^* is the potential intention o that satisfies the intention coherence constraint.

This is a constraint on action evolution learning in the intention space. First, we need to determine whether the current potential intention o is closer to the real intention compared to the initial potential goal $\bar{o}$. This can gradually bring the current potential intention o closer to the real intention and promote the update of the best potential intention o^*, as shown below

$$\|o^* - o\|_2^2 > \|\bar{o} - o\|_2^2 . \tag{7}$$

In addition, we set a threshold ξ to measure the distance between the optimal potential intention o^* and the current potential intention o to ensure that the change in the current potential intention o is not too outrageous, as shown below

$$\|o^* - o\|_2^2 \leq \xi. \tag{8}$$

Finally, we perform full connection FC and linear regression on the optimal latent intention o^* respectively to obtain the evolutionary action $\hat{y}' \in R^{\mathbb{B}\times(\mathbb{C}+1)}$ and evolution time $\tilde{D} \in R^{\mathbb{B}\times 1}$.

Loss function. The loss consists of four parts: action evolution loss L_{AE}, potential action loss L_{PA}, evolution timing loss L_{ET} and intention coherence loss$L_{coherence}$.

The action evolution loss L_{AE} aims to constrain the consistency between the predicted evolution category $\hat{y}'$ and the next category of the ground truth y'. We use our designed ICCA performance indicator in conjunction with cross-entropy loss calculation as follows,

$$L_{AE} = ICCA\_loss_{ce}(\hat{y}', y'). \tag{9}$$

The potential action loss L_{PA} aims to constrain the consistency between the initialized recognition of the evolution category $\hat{y}$ and the current category of the ground truth y. The representation is the same as Equation 9.

We perform L1 loss for the evolution timing loss L_{ET},

$$L_{ET} = \left\| \tilde{D} - D \right\|_1, \tag{10}$$

where D is the ground truth of the current action duration based on the sliding window.

Intention coherence loss $L_{coherence}$ consists of two parts: update loss L_u and maintenance loss L_m. Update loss L_u forces the current potential intention o to be closer to the real intention than the initial potential intention $\bar{o}$,

$$L_u = max(0, \|o^* - o\|_2^2 - \|\bar{o} - o\|_2^2 + m), \tag{11}$$

where m is a very small value to ensure numerical stability.

Maintenance loss L_m ensures that the potential intention o is consistent during training and that the changes are not too outrageous. Using max-margin loss, the deviation between the threshold ξ and the difference between the potential intention is measured.

$$L_m = max(0, \|o^* - o\|_2^2 - \xi + m). \tag{12}$$

Therefore, $L_{coherence} = L_u + L_m$.

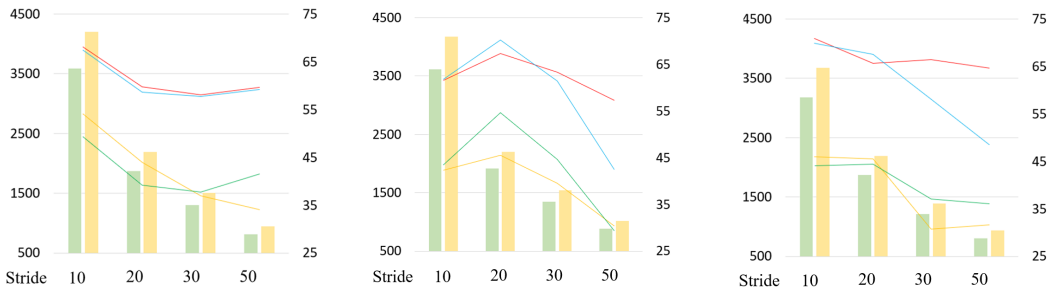
The overall loss can be represented as,

$$L = L_{AE} + L_{PA} + L_{ET} + L_{coherence}. \tag{13}$$

Experiment

Experimental Setup

Dataset. We use two popular untrimmed human action datasets, THUMOS14 (Idrees et al. 2017) and ActivityNET v1.3 (Caba Heilbron et al. 2015), as our benchmark datasets. The THUMOS14 dataset is a large-scale video dataset that includes 1,010 videos for validation and 1,574 videos for testing from 20 classes. Among all the videos, there are 220 and 212 videos with temporal annotations in validation and testing set, respectively. Following previous works (Wang et al. 2017; Paul, Roy, and Roy-Chowdhury 2018; Luo et al. 2020), we use the 200 videos in the validation set for training



(a) length=50 (b) length=100 (c) length=150

Figure 5: The effect of sliding window length and stride. The left y-axis represents sample size, the right y-axis represents accuracy (%), and the x-axis represents stride values of {10, 20, 30, 50}. Green bars denote training set samples, while yellow bars represent test set samples. The red and blue lines show the top-1 recognition accuracy and prediction accuracy. The yellow and green lines represent the mean precision on recognition and predictive. All experiments are based on the ICCA loss.

loss function	top-1 acc		top-5 acc		MP	
	reco	pred	reco	pred	reco	pred
focal loss	59.2	61.1	92.7	88.1	37.76	37.96
weighted-CE	57.1	59.1	90.4	87.3	23.91	23.95
EQLv1	65.2	59.6	87.4	74.6	39.73	37.12
EQLv2	57.8	59.3	89.1	90.9	30.62	36.28
CDB loss	66.1	65.8	92.7	**92.5**	39.26	50.10
ICCA loss	**67.3**	**70.2**	**92.9**	91.0	**45.47**	**54.61**

Table 1: Performance results obtained on the THUMOS14 dataset using different loss functions. Where $reco$ denotes recognition, $pred$ denotes prediction, acc denotes accuracy, and MP denotes mean precision.

loss function	top-1 acc		top-5 acc		MP	
	reco	pred	reco	pred	reco	pred
focal loss	59.3	16.0	75.8	38.9	4.16	7.77
weighted CE	63.9	**36.1**	83.9	64.8	13.76	26.54
EQLv1	58.8	30.3	72.3	44.2	12.05	23.09
EQLv2	64.8	**36.1**	72.4	61.0	16.63	28.75
CDB loss	64.3	28.9	81.9	54.6	10.20	20.14
ICCA loss	**65.0**	34.3	**84.0**	**63.5**	**18.65**	**29.47**

Table 2: Performance results obtained on the ActivityNET v1.3 dataset using different loss functions.

and the 213 videos in the testing set for evaluation. ActivityNET v1.3 is a large-scale dataset with 200 complex daily activities. It has 10,024 training videos and 4,926 validation videos. Following (Yang et al. 2021; Luo et al. 2021), we use the training set as current data stream to train our model and the validation set as new data stream for evaluation.

Metrics. We use the average accuracy (top1, top5) and the mean precision (MP) of each class as evaluation for long-tail distribution. The former reflects the overall evaluation performance, while the latter can accurately evaluate the weight correction effect of our ICCA loss on head and tail classes. For untrimmed intent estimation task, in addition to using the average accuracy (top1, top5) , we use time accuracy to evaluate the estimation of evolution duration (following (Rodin et al. 2022)), where time accuracy is defined as the

dataset	Methods (trimmed)	top-1 acc		top-5 acc	
		reco	pred	reco	pred
THUMOS14	RULSTM	– –	50.3	– –	67.0
	latent goal	58.7	**54.8**	94.7	92.1
	ours	**59.7**	54.3	**95.0**	**93.1**
ActivityNET v1.3	RULSTM	– –	35.7	– –	78.5
	latent goal	49.0	39.9	79.3	85.0
	ours	**55.8**	**45.7**	**82.3**	**88.1**

Table 3: Recognition and prediction results using different backbones on the trimmed THUMOS14 and ActivityNET v1.3 datasets. For fair comparison, our trimmed sample sets include the "background" class.

percentage of samples whose predicted duration is within 1 second of the ground truth duration.

Implementation details. Details of experimental parameters are provided in the supplementary material.

Experimental Results

Comparison with various long-tail distribution loss functions. We compare popular loss functions with adjusted weights on the THUMOS14 and ActivityNET v1.3 datasets (Table 1 and 2). Focal loss (Lin et al. 2017) increases the weight of majority classes by intuition at the beginning of training, without considering the problem that though the "background class" in untrimmed video samples is a majority class, it is difficult to classify it as one class. Weighted-CE, EQLv1 (Tan et al. 2020) and EQLv2 (Tan et al. 2021) use hard weights based on the training set sample distribution as a reference, ignoring the problem of overfitting on the training set when the difference between the training set and testing set distributions is too large. CDB loss (Sinha, Ohashi, and Nakamura 2022) uses a class-balanced subset to dynamically correct the weight distribution, to some extent correcting the overfitting problem on the training set, but without dynamically judging the data drift of training set and testing set samples. Considering that in the process of untrimmed video prediction, the long-tail problem may change in different class distributions and the difference be-

dataset	Methods (untrimmed)	top-1 acc reco	top-1 acc pred	top-5 acc reco	top-5 acc pred	time acc
THUMOS 14	RU-reg	−−	59.2	−−	92.6	35.30
	latent goal	60.1	56.8	92.6	89.8	37.76
	ours	**67.9**	**64.8**	**95.5**	**92.8**	**38.11**
ActivityNET v1.3	RU-reg	−−	28.9	−−	55.7	**30.87**
	latent goal	63.5	31.0	81.5	58.5	14.50
	ours	**65.0**	**34.3**	**84.0**	**66.5**	27.04

Table 4: Recognition and prediction results and time estimation results using different backbones on the untrimmed THUMOS14 and ActivityNET v1.3 datasets

tween training set and test set distributions may be very large, we propose ICCA loss to specifically address the self-adaptive problem of long-tail distribution with drift characteristics in video datasets. Our method achieves optimal or suboptimal performance in mean accuracy (top-1, top-5) and mean precision of each class for recognition and prediction.

Comparison with trimmed methods. To be fair, samples are processed into video clips containing "background" to study the effectiveness of our backbone in the long tail distribution between classes (between action classes and background classes) and within classes (between background classes and background classes). RULSTM (Furnari and Farinella 2020) is designed to predict categories within 1 second in the future. Latent goal (Roy and Fernando 2022) recognizes and predicts the current category and the next category. These two methods are representative methods for predicting human actions on trimmed datasets. As can be seen from Table 3, our methods have achieved good performance.

Comparison with untrimmed methods. Table 4 shows the recognition and prediction results and time duration estimation results of different backbone on the untrimmed THUMOS14 and ActivityNET v1.3 datasets. Among them, RU-reg (Rodin et al. 2022) is aimed to predict the time-to-action by exploiting an additional fully connected layer attached to the RULSTM model and trained to solve the regression task and multi-classification task. In addition, latent goal is commonly used for multi-classification tasks. We also add a fully connected layer at the end of the model to perform a regression task. Table 4 shows that we achieved advanced results on both the two untrimmed datasets.

Ablation Experiment

The effect of sliding window length and stride. We have explored the impact of different lengths and strides of the sliding window. Experiments have found that the smaller the window length, the larger the sample size; with the same window length, the smaller the stride, the larger the sample size, and the mean precision is often larger. Taking into account the calculation cost and performance results, we chose length=100 and stride=20 as our experimental settings.

The effect of fusion frequency. We explored the impact of fusion frequency in ICCA loss (Table 5). We believe that fusion frequency is an important parameter to determine the frequency of new data stream and current data stream distribution calibration. Table 5 shows that the smaller the fusion

fusion frequency	top-1 acc reco	top-1 acc pred	top acc reco	top acc pred	MP reco	MP pred
5	66.3	64.7	90.4	**92.4**	40.48	45.42
10	**67.3**	**70.2**	92.9	91.0	45.47	**54.61**
20	64.7	61.6	**93.1**	90.8	**48.05**	50.31
30	63.7	60.1	89.6	91.0	37.08	39.19

Table 5: The effect of fusion frequency.

weight strategy	error function	top-1 acc reco	top-1 acc pred	top-5 acc reco	top-5 acc pred	MP reco	MP pred
sum	each	66.0	65.0	92.6	89.6	43.86	47.33
mean	class	**67.3**	57.9	94.3	88.6	34.06	37.20
max	precision	**67.3**	**70.2**	92.9	**91.0**	**45.47**	**54.61**
	f1-score	65.2	63.8	92.8	89.7	30.4	40.67
	recall	67.3	59.9	**94.8**	88.9	33.40	36.82
	Gmean	64.1	57.3	94.0	90.6	32.84	38.11

Table 6: The effect of error function and weight strategy.

frequency are, the more frequently the current data stream and new data stream are distributed for calibration, and the predicted results tend to be more accurate. However, if the fusion frequency is set to 5, the frequency is too fast, which is not conducive to the convergence of the training model, and the calculation burden will be increased to some extent, so the performance is not the best. Based on the above considerations, we selected fusion frequency=10 as our experimental setting.

The effect of error function. We have explored the impact of various error functions on ICCA loss (Table 6). The error function is a discussion of the error function $\zeta(c_j)$. We have designed four strategies: f1-score , class precision, recall, and Gmean. Experiments have found that using mean precision performs best.

The effect of weight strategy. We have explored the impact of weight strategy on ICCA loss (Table 6). The weight strategy is the implementation of the Ω function in Equation 4. We have designed three strategies: sum, mean, and max. Experiments have found that using the max strategy performs best.

Conclusion

We designed a method to predict human intentions in untrimmed videos based on intentional evolutionary learning. Specifically, an ICCA loss is presented to alleviate prediction bias caused by long-tail distribution with concept drift characteristics. Moreover, an intention-oriented evolutionary learning method is proposed to determines the intention evolution patterns and the time of evolution. Extensive experiments show that our method can achieve better results on untrimmed video than fine-tuned trimmed methods. While the paper presents an innovative approach to intention pattern detection, there are opportunities for further improvement. By improving the accuracy of the analysis models for the time of intention evolution, future research can advance the field of human intention understanding.

Acknowledgments

The authors gratefully acknowledge the financial supports by the National Natural Science Foundation of China under Grant 92048301 and Grant 62202332, and Diversified Investment Foundation of Tianjin under Grant 21JCQNJC00980.

References

Abu Farha, Y.; Richard, A.; and Gall, J. 2018. When will you do what?-anticipating temporal occurrences of activities. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5343–5352.

Bai, L.; Yao, L.; Li, C.; Wang, X.; and Wang, C. 2020. Adaptive graph convolutional recurrent network for traffic forecasting. *Advances in neural information processing systems*, 33: 17804–17815.

Bodenstedt, S.; Wagner, M.; Mündermann, L.; Kenngott, H.; Müller-Stich, B.; Breucha, M.; Mees, S. T.; Weitz, J.; and Speidel, S. 2019. Prediction of laparoscopic procedure duration using unlabeled, multimodal sensor data. *International Journal of Computer Assisted Radiology and Surgery*, 14: 1089–1095.

Caba Heilbron, F.; Escorcia, V.; Ghanem, B.; and Carlos Niebles, J. 2015. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 961–970.

Carreira, J.; and Zisserman, A. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6299–6308.

Furnari, A.; and Farinella, G. M. 2020. Rolling-unrolling lstms for action anticipation from first-person video. *IEEE transactions on pattern analysis and machine intelligence*, 43(11): 4021–4036.

Gao, J.; Yang, Z.; and Nevatia, R. 2017. Red: Reinforced encoder-decoder networks for action anticipation. *arXiv preprint arXiv:1707.04818*.

Girdhar, R.; and Grauman, K. 2021. Anticipative video transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, 13505–13515.

Grigorev, A.; Mihaita, A.-S.; Lee, S.; and Chen, F. 2022. Incident duration prediction using a bi-level machine learning framework with outlier removal and intra–extra joint optimisation. *Transportation research part C: emerging technologies*, 141: 103721.

He, B.; Yang, X.; Kang, L.; Cheng, Z.; Zhou, X.; and Shrivastava, A. 2022. ASM-Loc: action-aware segment modeling for weakly-supervised temporal action localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13925–13935.

Idrees, H.; Zamir, A. R.; Jiang, Y.-G.; Gorban, A.; Laptev, I.; Sukthankar, R.; and Shah, M. 2017. The thumos challenge on action recognition for videos "in the wild". *Computer Vision and Image Understanding*, 155: 1–23.

Ke, Q.; Fritz, M.; and Schiele, B. 2019. Time-conditioned action anticipation in one shot. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9925–9934.

Krawczyk, B.; Minku, L. L.; Gama, J.; Stefanowski, J.; and Woźniak, M. 2017. Ensemble learning for data stream analysis: A survey. *Information Fusion*, 37: 132–156.

Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollár, P. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, 2980–2988.

Liu, B.; Chen, Y.; Liu, S.; and Kim, H.-S. 2021. Deep learning in latent space for video prediction and compression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 701–710.

Luo, W.; Zhang, T.; Yang, W.; Liu, J.; Mei, T.; Wu, F.; and Zhang, Y. 2021. Action unit memory network for weakly supervised temporal action localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9969–9979.

Luo, Z.; Guillory, D.; Shi, B.; Ke, W.; Wan, F.; Darrell, T.; and Xu, H. 2020. Weakly-supervised action localization with expectation-maximization multi-instance learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16*, 729–745. Springer.

Paul, S.; Roy, S.; and Roy-Chowdhury, A. K. 2018. W-talc: Weakly-supervised temporal activity localization and classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 563–579.

Rodin, I.; Furnari, A.; Mavroeidis, D.; and Farinella, G. M. 2022. Untrimmed action anticipation. In *Image Analysis and Processing–ICIAP 2022: 21st International Conference, Lecce, Italy, May 23–27, 2022, Proceedings, Part III*, 337–348. Springer.

Roy, D.; and Fernando, B. 2022. Action anticipation using latent goal learning. In *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV 2022, Waikoloa, HI, USA, January 3-8, 2022*, 808–816. IEEE.

Sinha, S.; Ohashi, H.; and Nakamura, K. 2022. Class-Difficulty Based Methods for Long-Tailed Visual Recognition. *International Journal of Computer Vision*, 130(10): 2517–2531.

Tan, J.; Lu, X.; Zhang, G.; Yin, C.; and Li, Q. 2021. Equalization loss v2: A new gradient balance approach for long-tailed object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1685–1694.

Tan, J.; Wang, C.; Li, B.; Li, Q.; Ouyang, W.; Yin, C.; and Yan, J. 2020. Equalization loss for long-tailed object recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11662–11671.

Wang, L.; Xiong, Y.; Lin, D.; and Van Gool, L. 2017. Untrimmednets for weakly supervised action recognition and detection. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 4325–4334.

Wang, W.; Peng, X.; Su, Y.; Qiao, Y.; and Cheng, J. 2021. Ttpp: Temporal transformer with progressive prediction for efficient action anticipation. *Neurocomputing*, 438: 270–279.

Wanyan, Y.; Yang, X.; Ma, X.; and Xu, C. 2023. Dual Scene Graph Convolutional Network for Motivation Prediction. *ACM Transactions on Multimedia Computing, Communications and Applications*, 19(3s): 1–23.

Yang, W.; Zhang, T.; Yu, X.; Qi, T.; Zhang, Y.; and Wu, F. 2021. Uncertainty guided collaborative training for weakly supervised temporal action detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 53–63.

Zheng, N.; Song, X.; Su, T.; Liu, W.; Yan, Y.; and Nie, L. 2023. Egocentric Early Action Prediction via Adversarial Knowledge Distillation. *ACM Transactions on Multimedia Computing, Communications and Applications*, 19(2): 1–21.