ANOMALY DETECTION WITH WEAKLY SUPERVISED LEARNING

Anonymous authors
Paper under double-blind review

1 REPRODUCIBILITY SUMMARY

1.1 SCOPE OF REPRODUCIBILITY

This work tries to reproduce the results of ICCV 2021 paper: 'Weakly-Supervised Video Anomaly Detection With Robust Temporal Feature Magnitude'. Paper suggests about novel method of weakly-supervised anomaly detection. We tries to reproduce the results in different constrains: loss, parameters, and data.

1.2 METHODOLOGY

Our GPU asset was rtx3090. We references author's open source code while changing the other parameter conditioning.

1.3 RESULTS

This work searched about different RTFM's experiment results. Results 1 shows the importance of RTFM's loss and effect of weighted cce loss. Results 2 shows the best number of k in RTFM. Results 3 shows the output of different method of gathering data. Results 4 shows the experiment of changing feature selection.

2 INTRODUCTION

In the Anomaly Detection area, there is issue that rare abnormal snippets in the abnormal videos is largely biased by the dominant negative instances. This can lead estimation of anomalies exhibit only small differences with normal videos. The issue became worse when detection algorithm ignores video temporal dependencies. Robust Temporal Feature Magnitude learning(RTFM) can address this issue. RTFM trains a feature magnitude learning function to effectively recognize positive instances and improve the robustness of the MIL approach to the negative instances from abnormal videos.

Traditionally, one-class classifiers, which is called unsupervised anomaly detection, is has been researched, but the best performing approach is weakly supervised anomaly detection. This method got optimal point of human resources and detection's performance. And the next challenge is solving major problems of weakly supervised anomaly detection. MIL algorithm can mitigates these problems but still exists.

There is four problems while using MIL in weakly-supervised anomaly detection. 1) the top anomaly score in an abnormal video may not be from an abnormal snippet; 2) normal snippets randomly selected from normal videos may be relatively easy to fit, which challenges training convergence; 3) if the video has more than one abnormal snippet, we miss the chance of having a more effective training process containing more abnormal snippets per video; 4) the use of classification score provides a weak training signal that does not necessarily enable a good separation between normal and abnormal snippets.

In the face of these problems, author Yu uses robust temporal feature magnitude learning to improve robustness of MIL. Our proposed RTFM receives a TxD feature matrix F extracted from a video containing T snippets. Then Multi-scale Temporal Feature Learning captures the long and short-range temporal dependencies between snippet features to produce X matrix. Next, we maximise the

CamScanner로 스캔하기

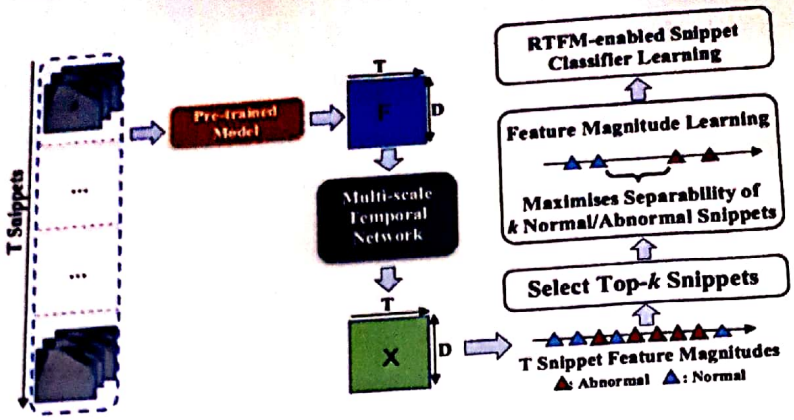

Figure 1: RTFM

separability between abnormal and normal video features and train a snippet classifier using the top-k largest magnitude feature snippets from abnormal and normal videos. Detail method illustration attached in Figure 1.

3 METHODOLOGY

We re-implemented the experiment of the paper using the open source provided by author. Because of limitation of time constraints, we made experiment on one data set: ShanghaiTech dataset.

3.1 MODEL DESCRIPTIONS

We used the extracted videos features which were extracted by pre-trained i3d video feature extractor, and the final features were extracted using the multi-scale temporal network proposed in this paper. The network extracts the degree of anomality, and an abnormal one is represented by 1 and a normal one by 0. The network extracts features containing temporal and spatial characteristics and expresses video in one score through 3 fully connected layers. Due to a few data used in the paper, the structure of the simple network suggested was used. We predicted that the performance would degrade by overfitting if the model becomes deeper.

3.2 DATASETS

In this paper, experiments were conducted using UCF-Crime, XD-Violence, ShanghaiTech, and UCSD-Peds dataset, and among them, we selected the ShanghaiTech dataset, which has a relatively small amount. There are a total of 238 clips for training, 63 abnormal and 175 normal, and the test set has 199 clips. The total frame of the test set is 142912, of which 8028 are abnormal frames. The total number of videos in train and test set is 13. Each video is divided into multiple clips, and each clip is composed of multiple frames. Train set was given a label in units of clips, and the normal frame may be incorrectly labeled as abnormal in the normal clip. The test set is given a label in frame units, and there is no wrong label.

3.3 HYPERPARAMETERS

Early stopping point was set to 10 for train loss, and the max iteration was 15,000. The batch size was 32, and Adam optimizer with weight decay 0.005 was used, and the learning rate was set to 0.001. Normal 32 clips and abnormal 32 clips are randomly selected for each minibatch.

We changed the method of calculating loss to a total of four ways. In Experiment 1, we changed the RTFM loss itself to weighted cce loss of the recently proposed BARE algorithm used for weakly supervised learning. The method of extracting and using the top three features is fixed.

CamScanner로 스캔하기

In Experiment 2, the value of selecting the top k features was changed, and we used the RTFM loss.

In Experiment 3, a method of calculating the magnitude between features was added.

In Experiment 4, the method of selecting representative feature values was changed. When calculating the loss, only one top was selected.

3.4 Experimental setup and code

In Experiment 1, RTFM loss using top 3 features of each of normal and abnormal was used, and weighted cce loss was added to RTFM loss. To check the performance of weighted cce loss itself, RTFM loss was removed and experiment was conducted using only weighted cce loss. The sparse loss to reduce the calculation and the smooth loss to reduce the difference between the extracted feature were used.

In Experiment 2, each experiment was conducted by changing the hyperparameter to use the upper 1, 3 (base), 5, 10, and all features.

In Experiment 3, the experiment was conducted using only the addition of the loss that reduces the standard deviation of each feature to the smooth loss and the loss that reduces the standard deviation except for the smooth loss.

In Experiment 4, loss using only the top one and loss using a median were compared.

3.5 Computational requirements

The experiment was conducted using gpu3090. The maximum iteration was set to 15,000, but the training was saturated around 100 iteration, and it took about 5 minutes and 3 sec/iter to perform one experiment.

4 Result

Through 4 experiments, the following results were obtained. In the first experiment, the newly proposed weighted cce is not suitable for this problem. In the second experiment, determining the number of representative features has an important effect on performance. Previously, only three were used, but this number may be a small amount to represent the data in the batch. In the third experiment, it can be seen that the existing smooth loss alone works effectively to make the space where each normal and abnormal data are concentrated. However, it was confirmed that using the standard deviation can play a similar role. Experiment 4 derives the fact that using the middle value rather than using only the top one can represent the distribution of the data well.

There was an improvement in performance according to how to set the representative data through Experiment 2. It is expected that good performance would be improved if features that fall within the range is set as representative values using the weight cce loss of Experiment 1 and the standard deviation. We leave in later experiments.

4.1 Results reproducing original paper

4.1.1 Result 1

Table 1: Change the method of calculating loss.

loss	AUC
RTFM(base)	0.9121
Weighted cce	0.5405
RTFM+Weighted cce	0.6843

If the wrong label is in training data, the characteristic appears in the distribution of the loss value. By calculating the average and standard deviation of the feature values, weighted cce can notice data

CamScanner로 스캔하기

that enters a specific range as accurately labeled data, and train by giving weight to data within the range. In the case of this experimental data, it worked well when the experiment was conducted after deliberately mis-labeling the mnist or cifar data, but in the case of Shanghai dataset, it would be difficult to extract common characteristics between abnormal data. Most of the wrong labels appear in normal data, and there is no wrong label in abnormal data. This problem includes weakly supervised learning, but the problem of anomaly detection, which has a small number of abnormal data, should be solved together, so abnormal data should be trained as much as possible. As a result of separately extracting and checking, there was no consistent characteristic between the features of abnormal data, so only a very small number of abnormal data were used for training, and as a result, the following results were found because training did not proceed properly.

4.1.2 RESULT 2

Table 2: Change the number of k

top k	AUC
top 3(base)	0.9121
top 1	0.8600
top 5	**0.9266**
top 10	0.9159
All minibatch features	0.9046

Through Experiment 1, data that well represent the characteristics of each class could be extracted, and each batch the data was extracted in various numbers from 3 to as many as 10. The amount of this extracted data is all different, and we expected that using more data would be helpful for training than setting values for only the top three representative data. As a result, when calculating loss by extracting five, we extract a little more representative data, which makes more general features available, and as a result, there was performance improvement.

4.1.3 RESULT 3

Table 3: Change the method of gathering data. In this experiment, we use total features for RTFM loss.

loss	AUC
smooth	0.9046
std	0.9018
smooth+std	0.8980

Looking at the distribution of the data, we predicted that the more normal and abnormal data gather each other, the more accurate the model will be able to distinguish between normal and abnormal data, so we experimented to reduce the standard deviation. In the existing smooth loss, when each data is listed in order, training in the direction of reducing the difference between each data plays a similar role, and as a result, it is confirmed that there is little difference in performance.

4.1.4 RESULT 4

Table 4: Change the method of selecting the feature. In this experiment, we use one features for RTFM loss.

method	AUC
top 1	0.8600
median	0.8828

CamScanner로 스캔하기

In the last experiment, we expected that using the values in the middle would cover the entire distribution more, and compared the results of using only the top one with the median values. As a result, it was confirmed that the performance was higher when one median was used.

5 DISCUSSION

This challenge started from novel anomaly detection method, RTFM. The method enables top-k MIL approaches for weakly supervised video anomaly detection. So it takes large margin between normal and abnormal snippets and easily detects the abnormal snippets which is blended with many normal snippets. As Results, RTFM able to achieve improved performance comparing other weakly-supervised anomaly detection.
Our experiments shows the results about different RTFM losses and parameters. We changed the calculating loss to know about effect of weighted cce loss.If we separate weighted cce and RTFM loss, only small number for abnormal data was used so model didn't train well. About number of k, the number 5 was got bet score while we changed the number of k. When we changed the method of gathering data, which reduce the standard deviation, it shows little difference in performance. Last experiment shows selecting the median value feature got higher performance than the top1 value.

REFERENCES

[1] Yoshua Bengio and Yann LeCun. Scaling learning algorithms towards AI. In *Large Scale Kernel Machines*. MIT Press, 2007.

[2] Geoffrey E. Hinton, Simon Osindero, and Yee Whye Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18:1527–1554, 2006.

[3] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT Press, 2016.

[4] Yu Tian, Guansong Pang, Yuanhong Chen, Rajvinder Singh, Johan W Verjans, and Gustavo Carneiro. Weakly-supervised video anomaly detection with robust temporal feature magnitude learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4975–4986, 2021.

[5] Deep Patel and PS Sastry. Adaptive sample selection for robust learning under label noise. *arXiv preprint arXiv:2106.15292*, 2021.

CamScanner로 스캔하기